Development and Implementation of a Method for Solving the Laminar Boundary Layer Equations in Airfoil Flows

Master-Thesis von Michael Köhler
August 2011

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Development and Implementation of a Method for Solving the Laminar Boundary Layer Equations in Airfoil Flows

Vorgelegte Master-Thesis von Michael Köhler

Betreuer:
Dipl-Ing. Andreas Reeh
Dipl-Ing. Alexander Duchmann

Tag der Einreichung:

Development and Implementation of a Method for Solving the Laminar Boundary Layer Equations in Airfoil Flows

TECHNISCHE UNIVERSITÄT DARMSTADT

Project definition for the Master's Thesis of B.Sc. Michael Köhler
Supervisors: Dipl-Ing. Andreas Reeh, Dipl-Ing. Alexander Duchmann
Start date: 07.02.2011

SLA

Formulation of the problem

Current investigations focus on the influence of atmospheric turbulence on natural laminar flow (NLF) airfoils under free-flight conditions. The turbulent disturbances affect the flow around the wing and may lead to a reduction in the aerodynamic performance of sailplanes with NLF wings. Potential mechanisms for this loss are increased friction drag due to premature transition (receptivity of the boundary layer), quasi-steady losses in the nonlinear region of the lift curve and changes through unsteady airfoil effects. The only possibility to capture all effects simultaneously, to test their relevance under different conditions and to investigate their mutual interactions is free-flight experiments.

In order to gain deeper insight into the mechanisms of action an aerodynamic model is developed. The model couples the solutions of sub-problems and allows the comparison with the experimental results. For the theoretical modeling of the problem a two-dimensional, unsteady panel method is coupled with boundary-layer computations in order to investigate the changes of linear stability of the airfoil boundary layer and the transition location. The detailed boundary-layer information provided by such computations is extremely important for additional research activities on transition control by means of plasma actuators.

Description of the task

The major task is the development of a method for solving the laminar, stationary, incompressible boundary-layer equations, which may be used as an independent solver or as an integral part of the complete model. The method should compute the boundary-layer evolution for prescribed pressure distributions. First of all, an integral boundary-layer method is to be implemented. The analytic approximation broadens the understanding for the mathematical nature of the physical problem and allows an efficient comparison to a computationally more expensive but more accurate numerical method, which is to be subsequently developed and implemented. The discretization of the boundary value problem should predict laminar separation points and provide the boundary-layer profile evolution as starting point for a separate linear stability analysis. The solutions of the boundary-layer methods are to be validated and verified by using self-similar solutions of the boundary-layer equations and computations of the program XFOIL for different airfoil flows. A simple possibility for the coupling of the boundary-layer methods to the other model components is to be considered.

The possibility of an unsteady development of the boundary layer should be evaluated in the relevant literature. If a quasi-steady consideration of the boundary layer is not appropriate for modeling the entire parameter range of the flow problem suggestions should be made for an extension or a change of the boundary-layer method to incorporate the unsteady term in the equation of motion.

Prof. Dr.-Ing. C. Tropea
Technische Universität Darmstadt
Fachgebiet Strömungslehre und Aerodynamik
Petersenstraße 30, 64287 Darmstadt

Erklärung zur Master-Thesis

Hiermit versichere ich, die vorliegende Master-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 10. August 2011

(Michael Köhler)

Abstract

This work describes the implementation of different methods to solve the laminar boundary-layer equations by using integral methods and a finite difference scheme. The implementation is carried out for a direct discretization of the boundary-layer equation and a more complex technique based on the Keller box. Further, different boundary-layer transformations are investigated and implemented. The solvers are applied to a natural laminar flow airfoil to compute the boundary-layer characteristics and profiles. As the laminar boundary-layer calculation diverges at the separation point a zero-equation turbulence model is implemented in the laminar code and an inverse method is derived. Both methods can calculate the boundary-layer characteristics and profiles for the whole airfoil. In the second part of the work transition prediction based on the calculated velocity profiles is investigated more closely. The results of a local stability solver are compared to the transition prediction results of a stability analysis based on parabolized stability equations.

Acknowledgements

I want to thank Prof. Dr.-Ing. Cameron Tropea for making this work possible and my supervisors Andreas Reeh and Alexander Duchmann for their academic support during my thesis. During the countless discussions with them I broadened my thinking about several aspects and new ideas came to my mind which finally led to the outcome of this work. Further, I want to thank my advisors and the Center of Smart Interfaces for making the visit at KTH Stockholm possible. Many results of my work would not have been possible without the exchange of ideas and codes with the *Linné Flow Centre* at KTH Stockholm. Especially, I would like to acknowledge Dr. Ardeshir Hanifi and David Tempelmann for their time and patience during many fruitful conversations and the great organization which led to a very pleasant stay in Stockholm. Thanks also to Andreas Güttler for having me introduced to my current advisors and thus enabling this thesis. Last but not least, I like to acknowledge all the employees and my fellow students in Griesheim for their support and the great working atmosphere.
Apart from the academic support, I like to thank my family, my girlfriend and all my other friends for their continuous encouragement during my studies in Darmstadt, Stockholm and Blacksburg and especially for their support during the last few months.

Contents

1. Introduction **1**

2. Theoretical Background **3**

2.1. The Navier-Stokes Equation . . . 3
2.2. Derivation of the Boundary-Layer Equations . . . 5
2.3. Integral Boundary-Layer Equation . . . 8
2.4. Solutions of the Integral Boundary-Layer Equation . . . 10
2.4.1. Pohlhausen Method . . . 10
2.4.2. First Walz Method . . . 13
2.4.3. Second Walz Method . . . 15
2.4.4. Thwaites Method . . . 16
2.5. Transformations of the Boundary-Layer Equation . . . 18
2.5.1. The Falkner-Skan Transformation . . . 18
2.5.2. Görtler Transformation . . . 22
2.6. Turbulence Models . . . 25
2.6.1. Cebeci and Smith Turbulence Model . . . 25
2.7. Inverse Boundary-Layer Method . . . 27
2.7.1. Method of Catherall and Mangler . . . 27
2.7.2. Method of Veldman . . . 27
2.8. Transition Prediction by Means of Stability Analysis . . . 29
2.8.1. Transition Prediction by Local Stability Analysis . . . 30
2.8.2. Transition Prediction by Parabolized Stability Equations . . . 33

3. Implementation of Laminar Boundary-Layer Methods **35**

3.1. Numerical Considerations . . . 35
3.1.1. The Boundary-Layer Grid . . . 35
3.1.2. Finding the Stagnation Point . . . 38
3.1.3. Differentiation Operator . . . 39
3.1.4. Tridiagonal Matrix Solver - Thomas Algorithm . . . 43
3.2. Solutions of the Integral Boundary-Layer Equation . . . 45
3.2.1. Walz Method . . . 45
3.2.2. Thwaites Method . . . 46
3.3. Solution Method for Similarity Flows . . . 47
3.4. Discretization of the Laminar Boundary-Layer Equation for Airfoil Flows . . . 50
3.4.1. Discretization of the Boundary-Layer Equation using the Direct Method . . . 51
3.4.2. Discretization of the Boundary-Layer Equation using the Keller Box Method . . . 59
3.4.3. Comparison of Boundary-Layer Calculation Methods . . . 77

4. Implementation of a Turbulence Model and an Inverse Boundary-Layer Method 79
4.1. Cebeci and Smith Turbulence Model 79
4.1.1. Direct Discretization Method 80
4.1.2. Keller Box Discretization Method 83
4.2. Implementation of the Inverse Boundary Layer 86

5. Implementation of Stability Analysis for Transition Prediction 92
5.1. The Chebyshev Discretization 92
5.2. Implementation of the Local Stability Solver 94
5.2.1. The Eigenvalue Solver 94
5.2.2. Computation of Growth Rate and eN-Method 96
5.2.3. Comparison with an Empirical Envelope Method 99
5.3. Implementation of a Solver for the Parabolized Stability Equations 101
5.3.1. Computation of Growth Rate and eN-Method 102

6. Conclusions and Outlook 105

A. Input File Format for Boundary-Layer Calculation 108

B. Output Format of the Boundary-Layer Codes 109

Nomenclature 110

List of Figures 117

Literature 119

1 Introduction

In the field of aerodynamics and in many other engineering disciplines the most important task is to increase efficiency to save resources. As airplanes are crucial in today's transportation sector it is important to investigate new ways to reduce drag and fuel consumption. Especially in times with steadily rising energy prices and ongoing changes to a more environmentally caring society aircraft industries have the chance to introduce new profile designs that fulfill those requirements. One highly investigated type is the "natural laminar flow" (NLF) airfoil that delays laminar-turbulent transition and thus significantly reduces drag for the design point. However, if flight conditions differ from the design point the transition location changes and the drag balance can turn the other way around. In order to gain a better understanding which flight conditions result in a positive or negative energy balance free flight experiments are performed. Thereby, the atmospheric turbulence levels are measured and resulting transition points are captured.

On the one hand free-flight experiments capture the actual physics in real flights and thus are the best reference data that can be obtained. On the other hand, those experiments are very time consuming and data cannot be reproduced as flight conditions will never be exactly the same during the period of measurement. In order to reduce the costs to design NLF airfoils and to derive models that predict the performance of such airfoils, numerical methods need to be investigated. In the following work a method will be derived to efficiently calculate the laminar boundary and the corresponding velocity profiles and their derivatives. Further, transition prediction methods are discussed and implemented which will capture the onset of transition. Thereby, the numerical methods can be used to understand the physics which lead to transition and how it can be delayed.

In Figure 1.1 the NLF airfoil is shown which is investigated in the following work. The airfoil is a modified Althaus AH93-157 which is very similar to the airfoil DU84-158 used on the glider ASW24. In the work the airfoil will be referred to as *MW-airfoil* as it was design by Michael Weismüller [28]. The chord length of the airfoil is $L_c^* = 1.35\,\mathrm{m}$ whereas the airfoil coordinates, x_c and y_c, refer to the horizontal and vertical coordinate of the airfoil. In the following work quantities with the superscript $(^*)$ are assumed to be given in dimensional values and variables without that superscript are assumed to be dimensionless. Further, the angle of attack α is shown which is counted positively in the sketched direction. In order to fully characterize the flow, the global Reynolds number has to be given which is based on the free-stream velocity at infinity (U_∞^*) and the chord length (L_c^*). The combination of the geometry of the airfoil, the global Reynolds number and the angle of attack is used to calculate a pressure distribution by an inviscid solver. In the following work the inviscid solver will be *XFOIL* if not explicitly stated differently. *XFOIL* calculates the pressure distribution with a panel method.

The following work will outline a numerical method that is capable of solving the steady, laminar, incompressible boundary-layer equations for a given pressure distribution. In the first step, the integral boundary-layer equations is solved that results only in characteristic boundary-layer values such as the displacement thickness. The methods presented here and not capable of computing the velocity profiles. In the next step, the boundary-layer equation are solved with efficient numerical schemes. Thereby, velocity profiles are calculated that fulfill the no-slip condition at the surface of the airfoil and asymptotically merge into the local free-stream velocity at the boundary-layer edge. The shape of those velocity profiles is important to predict the laminar-turbulent transition location and thus the boundary-layer

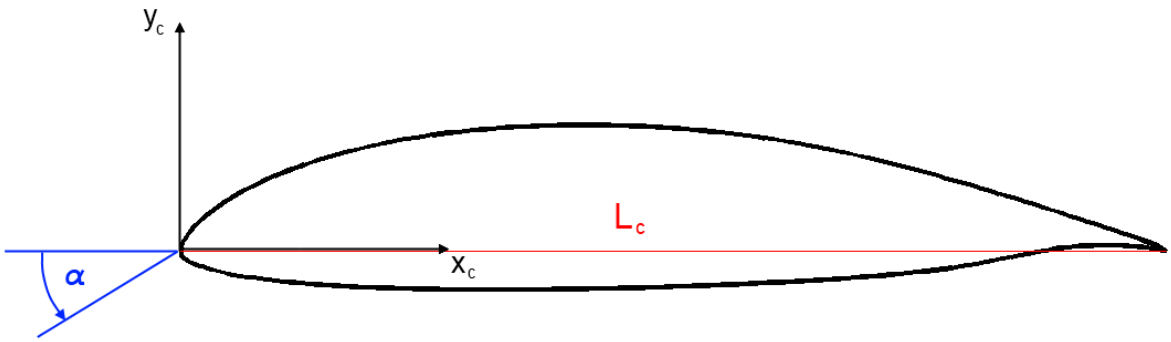


Figure 1.1.: The shape of the *MW-airfoil* and characteristic flow values.

solver is a necessary tool for a following stability analysis to predict laminar-turbulent transition.
The laminar boundary-layer equations are numerically solvable for positions where the boundary layer does not encounter separation. However, the laminar airfoil that is investigated as shown in Figure 1.1 consists of regions where separating flows occur and the laminar boundary-layer calculation diverges. Those regions are found at the dent at the end on the bottom of the airfoil and somewhere close to the middle at the top of the airfoil. The dent at the bottom side of the profile is necessary to obtain the momentum balance of the profile. The momentum balance of the profile must be suited to the remainder of the wing of the airplane. In order to overcome the problem, a zero-equation turbulence model is introduced and implemented in the boundary-layer code. The benefit of such a code is that the boundary layer can be calculated for the whole profile. The procedure to calculate the velocity profiles for a given pressure distribution is referred to as *standard method*.
As a final step a different calculation procedure is introduced and implemented that is also capable of iterating the edge velocity. Especially at the transition point the edge velocity dramatically changes and alters the pressure distribution of the airfoil. In order to avoid the update of the pressure distribution by an inviscid solver for each iteration step, a method is introduced that updates the edge velocity for a given displacement thickness distribution. Such a method, which manipulates the potential flow solution simultaneously, is referred to as *inverse method*. A very efficient method will be shown that iterates the displacement thickness and edge velocity quasi-simultaneously.
After having solved the boundary-layer equation, a stability analysis follows that is used for the prediction of the transition point. For such an analysis the solutions of the flow quantities are assumed to represent the mean flow and small perturbations are introduced that are investigated. Under specific frequencies and flow conditions those small disturbances can destabilize the mean flow and lead to laminar-turbulent transition. A simple but very robust method is shown first that includes only the wall-normal derivative of the streamwise mean flow. The flow is assumed locally parallel and the resulting Orr-Sommerfeld and Squire equations are solved.
A more elaborate stability analysis to predict laminar-turbulent transition is also introduced which takes into account the non-parallel effects of the mean flow as well. The resulting equations are known as the parabolized stability equations. The method is more complicated and numerically not as robust as the local stability solver. The benefit is a better agreement with real flows and thus a more realistic prediction of the transition point.

2 Theoretical Background

In the following chapter the theoretical background is explained and the important equations for the analysis of the boundary layer and the numerical implementation are derived. First of all, the Navier-Stokes equations are presented in section 2.1 which found the basis to derive the two-dimensional boundary-layer equation in direct and integral form in sections 2.2 and 2.3. Moreover, the three-dimensional Navier-Stokes equations are used to come up with the relevant stability equations to predict laminar-turbulent transition in airfoil flows in section 2.8.
Methods to solve the integral boundary-layer equation are presented in section 2.4 whereas well-known boundary-layer transformations to solve the direct boundary-layer equation are shown in section 2.5. The introduction and the theory of using turbulence models to compute the boundary layer for the whole airfoil is discussed in section 2.6. Finally, a more sophisticated method to compute the boundary layer by updating the edge velocity as well, is discussed in section 2.7.

2.1 The Navier-Stokes Equation

In general, the basic equations to describe a Newtonian flow in fluid dynamics are the continuity equation and the Navier-Stokes equations. In symbolic notation and for compressible flow the continuity equation is given by [20]

$$\frac{\partial \rho^*}{\partial t^*} + \nabla \cdot \left(\rho^* \vec{u^*}\right) = 0 \tag{2.1}$$

and the Navier-Stokes equations in their most general form are given by

$$\rho^* \frac{D\vec{u^*}}{Dt^*} = \vec{f^*} - \nabla p^* + \nabla \cdot \boldsymbol{\tau}^* \tag{2.2}$$

where ρ^* is the density of the fluid, $\vec{u^*}$ is the velocity vector consisting of $[u^*, v^*, w^*]^T$ for the velocity components in streamwise, normal and spanwise direction, $\vec{f^*}$ is the body force per unit volume taking into account for example the influence of gravity ($\vec{f^*} = \rho^* \vec{g^*}$), p^* is the pressure and $\boldsymbol{\tau}^*$ is the viscous stress tensor. The superscript (*) denotes that the quantities are given in dimensional form. The tensor $\boldsymbol{\tau}^*$ in cartesian form reads

$$\boldsymbol{\tau}^* = \begin{pmatrix} \tau^*_{xx} & \tau^*_{xy} & \tau^*_{xz} \\ \tau^*_{xy} & \tau^*_{yy} & \tau^*_{yz} \\ \tau^*_{xz} & \tau^*_{yz} & \tau^*_{zz} \end{pmatrix} \tag{2.3}$$

is symmetric and contains normal and deviatoric stresses. The derivation of those stresses in equation (2.3) can be found in [20]. By assuming the flow follows the laws of a Newtonian fluid the following assumptions can be made:

- There is a linear relation between the stress tensor and the rate of deformation
- The fluid is isotropic and thus there is no locally preferred direction

- In the hydrostatic stress state ($\vec{u^*} = 0$), all tangential forces vanish and thus $\nabla \cdot \boldsymbol{\tau}^* = 0$

The material law for Newtonian fluids is inserted for the stress tensor. If all the assumptions hold the components of the compressible Navier-Stokes equations for Newtonian fluids are given by

$$\rho^* \left(\frac{\partial u^*}{\partial t^*} + u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} + w^* \frac{\partial u^*}{\partial z^*} \right) = f_x^* - \frac{\partial p^*}{\partial x^*} + \frac{\partial}{\partial x^*} \left[\mu^* \left(2 \frac{\partial u^*}{\partial x^*} - \frac{2}{3} \nabla \cdot \vec{u^*} \right) \right] + + \frac{\partial}{\partial y^*} \left[\mu^* \left(\frac{\partial u^*}{\partial y^*} + \frac{\partial v^*}{\partial x^*} \right) \right] + \frac{\partial}{\partial z^*} \left[\mu^* \left(\frac{\partial u^*}{\partial z^*} + \frac{\partial w^*}{\partial x^*} \right) \right] \tag{2.4a}$$

$$\rho^* \left(\frac{\partial v^*}{\partial t^*} + u^* \frac{\partial v^*}{\partial x^*} + v^* \frac{\partial v^*}{\partial y^*} + w^* \frac{\partial v^*}{\partial z^*} \right) = f_y^* - \frac{\partial p^*}{\partial y^*} + \frac{\partial}{\partial y^*} \left[\mu^* \left(2 \frac{\partial v^*}{\partial y^*} - \frac{2}{3} \nabla \cdot \vec{u^*} \right) \right] + + \frac{\partial}{\partial z^*} \left[\mu^* \left(\frac{\partial v^*}{\partial z^*} + \frac{\partial w^*}{\partial y^*} \right) \right] + \frac{\partial}{\partial x^*} \left[\mu^* \left(\frac{\partial u^*}{\partial y^*} + \frac{\partial v^*}{\partial x^*} \right) \right] \tag{2.4b}$$

$$\rho^* \left(\frac{\partial w^*}{\partial t^*} + u^* \frac{\partial w^*}{\partial x^*} + v^* \frac{\partial w^*}{\partial y^*} + w^* \frac{\partial w^*}{\partial z^*} \right) = f_z^* - \frac{\partial p^*}{\partial z^*} + \frac{\partial}{\partial z^*} \left[\mu^* \left(2 \frac{\partial w^*}{\partial z^*} - \frac{2}{3} \nabla \cdot \vec{u^*} \right) \right] + + \frac{\partial}{\partial x^*} \left[\mu^* \left(\frac{\partial w^*}{\partial x^*} + \frac{\partial u^*}{\partial z^*} \right) \right] + \frac{\partial}{\partial y^*} \left[\mu^* \left(\frac{\partial v^*}{\partial z^*} + \frac{\partial w^*}{\partial y^*} \right) \right] \tag{2.4c}$$

where μ^* is the dynamic viscosity. In equations (2.4) Stokes hypothesis is used, so that the bulk viscosity is assumed to be $\lambda^* = -\frac{2}{3}\mu^*$ [20]. The bulk viscosity λ^* appears in the viscous stress tensor (2.3) and the validity of Stokes hypothesis is confirmed by a large number of experiments [20].
The Navier-Stokes equations in the given form above are elliptical and no analytical solution is available. Further, even modern computers take a long time to solve those equations numerically due to the need of highly resolved meshes and very small time steps. Therefore, the Navier-Stokes equations have to be simplified in order to achieve solutions for individual, specific problems. By simplifying the equations the result is only valid in correspondence to the simplifications. In the following, an incompressible flow (ρ^* is constant) is investigated. Further, the influence of body forces (e.g. gravity) is neglected and the terms containing the bulk viscosity $\lambda^* = -\frac{2}{3}\mu^*$ vanishes because the divergence is zero for incompressible flows. With those reductions the incompressible continuity equation and the incompressible, three-dimensional Navier-Stokes equations read

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} + \frac{\partial w^*}{\partial z^*} = 0 \tag{2.5}$$

and

$$\rho^* \left(\frac{\partial u^*}{\partial t^*} + u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} + w^* \frac{\partial u^*}{\partial z^*} \right) = -\frac{\partial p^*}{\partial x^*} + \mu^* \left(\frac{\partial^2 u^*}{\partial x^{*2}} + \frac{\partial^2 u^*}{\partial y^{*2}} + \frac{\partial^2 u^*}{\partial z^{*2}} \right) \tag{2.6a}$$

$$\rho^* \left(\frac{\partial v^*}{\partial t^*} + u^* \frac{\partial v^*}{\partial x^*} + v^* \frac{\partial v^*}{\partial y^*} + w^* \frac{\partial v^*}{\partial z^*} \right) = -\frac{\partial p^*}{\partial y^*} + \mu^* \left(\frac{\partial^2 v^*}{\partial x^{*2}} + \frac{\partial^2 v^*}{\partial y^{*2}} + \frac{\partial^2 v^*}{\partial z^{*2}} \right) \tag{2.6b}$$

$$\rho^* \left(\frac{\partial w^*}{\partial t^*} + u^* \frac{\partial w^*}{\partial x^*} + v^* \frac{\partial w^*}{\partial y^*} + w^* \frac{\partial w^*}{\partial z^*} \right) = -\frac{\partial p^*}{\partial z^*} + \mu^* \left(\frac{\partial^2 w^*}{\partial x^{*2}} + \frac{\partial^2 w^*}{\partial y^{*2}} + \frac{\partial^2 w^*}{\partial z^{*2}} \right). \tag{2.6c}$$

The three-dimensional Navier-Stokes equations are needed to come up with the stability equations to

predict the transition point which is shown later in the chapter in section 2.8. In the following work, a two-dimensional and incompressible (ρ^* is constant) mean flow is investigated. Therefore, the continuity equation (2.5) and the Navier-Stokes equations (2.6) simplify further as $w^* = 0$ and reduce to

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} = 0 \tag{2.7}$$

and

$$\rho^* \left(\frac{\partial u^*}{\partial t^*} + u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} \right) = -\frac{\partial p^*}{\partial x^*} + \mu^* \left(\frac{\partial^2 u^*}{\partial x^{*2}} + \frac{\partial^2 u^*}{\partial y^{*2}} \right) \tag{2.8a}$$

$$\rho^* \left(\frac{\partial v^*}{\partial t^*} + u^* \frac{\partial v^*}{\partial x^*} + v^* \frac{\partial v^*}{\partial y^*} \right) = -\frac{\partial p^*}{\partial y^*} + \mu^* \left(\frac{\partial^2 v^*}{\partial x^{*2}} + \frac{\partial^2 v^*}{\partial y^{*2}} \right). \tag{2.8b}$$

Those equations will be utilized to derive the two-dimensional boundary-layer equation in the next section.

2.2 Derivation of the Boundary-Layer Equations

After having derived the continuity equation (2.7) and Navier-Stokes equations for a two-dimensional and incompressible flow (2.8), the results are used for further simplifications to derive the boundary-layer equations.
In 1904, Prandtl introduced the boundary-layer concept and found out that viscous effects are dominant only in a small layer [1]. In the first step, an estimation of the order of magnitude for each term in equations (2.7) and (2.8) is employed to gain a deeper insight which terms are negligible. As inertial and viscous forces have the same order of magnitude in the boundary layer it follows that

$$Re = \frac{\text{Inertial Forces}}{\text{Viscous Forces}} = \frac{\rho^* u^* \frac{\partial u^*}{\partial x^*}}{\mu^* \frac{\partial^2 u^*}{\partial y^{*2}}} = \frac{u^* \frac{\partial u^*}{\partial x^*}}{\nu^* \frac{\partial^2 u^*}{\partial y^{*2}}} \approx 1. \tag{2.9}$$

A typical length scale for the x-direction is the chord length L_c^* and in y-direction the boundary-layer thickness δ^* whereas the velocity scale is usually the free-stream velocity U_∞^*. By using those quantities in equation (2.9) the following relation

$$\frac{\delta^*}{L_c^*} \approx Re^{-1/2} \tag{2.10}$$

is obtained. The Reynolds number Re is given by

$$Re = \frac{U_{ref}^* L_{ref}^*}{\nu^*} = \frac{U_\infty^* L_c^*}{\nu^*} \tag{2.11}$$

In the following work, the chord length L_c^* is taken as reference length L_{ref}^* whereas the free-stream velocity at infinity U_∞^* is usually taken as reference velocity U_{ref}^*. Obviously, from equation (2.10) follows that the ratio of normal to streamwise distance is given by $\delta^* \ll L_c^*$. In other words, the boundary-layer thickness is small compared to the length measure in streamwise direction. In order to fulfill the continuity equation (2.7) the wall-normal velocity must have the order of magnitude

$$v^* \approx U_\infty^* Re^{-1/2}. \tag{2.12}$$

This relation is found by inserting the scaled quantities into the continuity equation. The same ratio as for the length scaling holds for the velocity scaling as well. Therefore, $v^* \ll u^*$ which means that the streamwise velocity is assumed to be much higher than the wall-normal velocity.
In the next step, the continuity equation (2.7) and the Navier-Stokes equations (2.8) are transformed by

$$x = \frac{x^*}{L_c^*}, \qquad y = \frac{y^*}{L_c^*}, \qquad u = \frac{u^*}{U_\infty^*}, \qquad v = \frac{v^*}{U_\infty^*}, \qquad p = \frac{p^*}{\rho^* {U_\infty^*}^2} \quad \text{and} \quad t = \frac{t^* U_\infty^*}{L_c^*} \tag{2.13}$$

to obtain dimensionless equations. A closer investigation of the magnitude of each fraction in (2.13) shows that x and u have the order of magnitude $\mathcal{O}(1)$ whereas the estimations (2.10) and (2.12) proof that the wall-normal distance y and velocity v have the same order of magnitude which is $\mathcal{O}(\delta^*)$.
By applying the transformations (2.13) to the continuity equation (2.7) and the two-dimensional Navier-Stokes equations (2.8), the following non-dimensional equations can be found:

$$\underbrace{\frac{\partial u}{\partial x}}_{1} + \underbrace{\frac{\partial v}{\partial y}}_{1} = 0 \tag{2.14}$$

$$\underbrace{\frac{\partial u}{\partial t}}_{1} + \underbrace{u}_{1}\underbrace{\frac{\partial u}{\partial x}}_{1} + \underbrace{v}_{\delta}\underbrace{\frac{\partial u}{\partial y}}_{\frac{1}{\delta}} = -\frac{\partial p}{\partial x} + \underbrace{\frac{1}{Re}}_{\delta^2}\left(\underbrace{\frac{\partial^2 u}{\partial x^2}}_{1} + \underbrace{\frac{\partial^2 u}{\partial y^2}}_{\frac{1}{\delta^2}}\right), \tag{2.15a}$$

$$\underbrace{\frac{\partial v}{\partial t}}_{\delta} + \underbrace{u}_{1}\underbrace{\frac{\partial v}{\partial x}}_{\delta} + \underbrace{v}_{\delta}\underbrace{\frac{\partial v}{\partial y}}_{1} = -\frac{\partial p}{\partial y} + \underbrace{\frac{1}{Re}}_{\delta^2}\left(\underbrace{\frac{\partial^2 v}{\partial x^2}}_{\delta} + \underbrace{\frac{\partial^2 v}{\partial y^2}}_{\frac{1}{\delta}}\right). \tag{2.15b}$$

The order of magnitude of the Reynolds number is $\mathcal{O}(\delta^2)$ as previously derived in estimation (2.10) which satisfies the requirement that at least one friction term in the x-momentum equation must not vanish. That assumption is in accordance with results from experiments that v^* is proportional to the square of δ^*.
After checking the orders of magnitude for each term of equations (2.14) and (2.15) the terms that are of the order $\mathcal{O}(\delta)$ or smaller can be dropped since δ is assumed to be very small. This is reasonable as those terms disappear when the Reynolds number approaches infinity. In airfoil flow the Reynolds number is usually very large and thus fits very well with that assumption.
The continuity equation in dimensionless form reads

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{2.16}$$

and the momentum equations in x- and y-direction simplify to

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \frac{1}{Re}\frac{\partial^2 u}{\partial y^2} \tag{2.17a}$$

$$0 = -\frac{\partial p}{\partial y}. \tag{2.17b}$$

A direct implication of equation (2.17b) is the independence of the pressure in wall-normal direction. That means that the pressure distribution is only a function of x. Therefore, the pressure distribution can be taken from the edge of the boundary layer as it does not change in y-direction. At the edge of the boundary layer viscous effects vanish and that means that the pressure distribution can be calculated by using potential solutions. Another implication of that observation is that the velocity at the edge of the boundary layer can be directly computed by solving the momentum equation in streamwise direction at the edge of the boundary layer. At the boundary-layer edge the derivatives in wall-normal direction disappear and the edge velocity distribution is given by

$$\frac{\partial U_e}{\partial t} + U_e \frac{\partial U_e}{\partial x} = -\frac{\partial p}{\partial x} \tag{2.18}$$

where all viscous terms cancel out.
The boundary conditions are no-slip at the wall and free-stream velocity at the boundary-layer edge that is given by potential flow

$$\begin{aligned} y = 0: &\qquad u(x,y,t) = 0 \quad \text{and} \quad v(x,y,t) = 0 \\ y \to \infty: &\qquad u(x,y,t) = U_e(x,t). \end{aligned} \tag{2.19}$$

Equations (2.16) to (2.19) are in the following referred to as the dimensionless, incompressible, two-dimensional boundary-layer equations. For a steady flow the first term on the left-hand side of equations (2.17a) and (2.18) vanishes and the boundary conditions (2.19) are not time-depended any more. With those assumptions equations (2.16) to (2.19) can be referred to as the dimensionless, steady, incompressible, two-dimensional boundary-layer equations.
Finally, the boundary-layer equations for a steady, two-dimensional, incompressible flow in physical coordinates are

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} = 0 \tag{2.20}$$

$$\frac{\partial u^*}{\partial t^*} + u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} = -\frac{1}{\rho^*} \frac{\partial p^*}{\partial x^*} + \nu^* \frac{\partial^2 u^*}{\partial y^{*2}} \tag{2.21}$$

where the pressure distribution can also be given in terms of the velocity distribution at the edge of the boundary layer as

$$-\frac{1}{\rho^*} \frac{\partial p^*}{\partial x^*} = \frac{\partial U_e^*}{\partial t^*} + U_e^* \frac{\partial U_e^*}{\partial x^*}. \tag{2.22}$$

The boundary conditions are no slip at the wall and potential flow at the edge of the boundary layer resulting in

$$\begin{aligned} y^* = 0: &\qquad u^* = 0 \quad \text{and} \quad v^* = 0 \\ y^* \to \infty: &\qquad u^* = U_e^*(x,t). \end{aligned} \tag{2.23}$$

By simplifying the Navier-Stokes equation to the boundary-layer equation the number of variables is reduced from three (u, v, p) to only two (u, v). Further, the elliptical character of the original Navier-Stokes equations (2.8) vanishes and a parabolic system is obtain instead. In other words, the effects of the solution only act downstream of the flow and that simplifies the solution dramatically. If only the

steady solution of the boundary-layer equations is needed, the first term of equation (2.21) vanishes. The boundary-layer equations for the steady case read

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} = 0 \tag{2.24}$$

$$u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} = -\frac{1}{\rho^*} \frac{dp^*}{dx^*} + \nu^* \frac{\partial^2 u^*}{\partial y^{*2}} \tag{2.25}$$

with

$$-\frac{1}{\rho^*} \frac{dp^*}{dx^*} = U_e^* \frac{dU_e^*}{dx^*} \tag{2.26}$$

and the boundary conditions

$$\begin{aligned} y^* = 0: &\qquad u^* = 0 \quad \text{and} \quad v^* = 0 \\ y^* \to \infty: &\qquad u^* = U_e^*(x). \end{aligned} \tag{2.27}$$

The streamwise and wall-normal coordinates x and y do not need to be transformed for more complex geometries in case if the boundary-layer thickness δ is much smaller than the radius of curvature R. In those cases the boundary-layer coordinate system as shown in Figure 2.1 is similar to a Cartesian coordinate system and the Navier-Stokes equations including the boundary-layer equations are still valid without transformation [22]. At the same time that means that curvature effects are neglected in the following as they are assumed to be small.

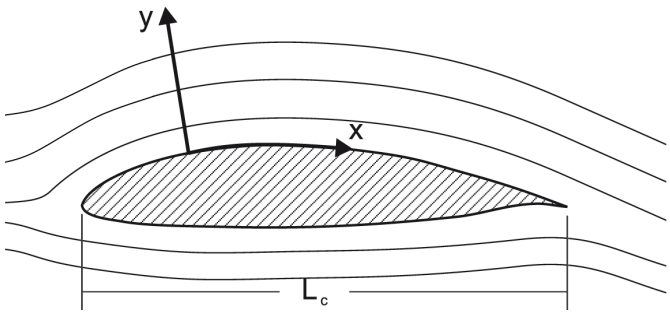


Figure 2.1.: Coordinate system in airfoil flows.

2.3 Integral Boundary-Layer Equation

The solution of the boundary-layer equation (2.25) simplifies if no velocity profiles are needed. In that case the integrated boundary-layer equation is used and it is solved only for the integral boundary-layer quantities. The displacement thickness δ_1^* is a measure for the thickness that has to be added on top of the airfoil if the solution would be computed inviscid. It is given by

$$\delta_1^* = \int_0^\infty \left(1 - \frac{u^*}{U_e^*}\right) dy^*. \tag{2.28}$$

The second integral boundary-layer quantity that is obtained is the momentum thickness δ_2^*. It represents the length that has to be added on top of the airfoil to obtain the same total momentum as for an inviscid flow. The momentum thickness is defined as

$$\delta_2^* = \int_0^\infty \left[\frac{u^*}{U_e^*} \left(1 - \frac{u^*}{U_e^*} \right) \right] dy^* \tag{2.29}$$

Those two quantities are used in the integral boundary-layer equation which will be derived in the following. The boundary-layer equation (2.25) is integrated with respect to y^* and then reads

$$\int_0^{h^*} \left(u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} - U_e^* \frac{dU_e^*}{dx^*} \right) dy^* = \int_0^{h^*} \nu^* \frac{\partial^2 u^*}{\partial y^{*2}} dy^* = \frac{\mu^*}{\rho^*} \left[\frac{\partial u^*}{\partial y^*} \right]_0^{h^*} = \frac{\tau_w^*}{\rho^*} \tag{2.30}$$

where h^* can be any position outside of the boundary layer and

$$\tau_w^*(x) = \mu^* \left. \frac{\partial u^*}{\partial y^*} \right|_{\text{wall}} \tag{2.31}$$

is the local shear stress evaluated at the wall. Outside of the boundary layer the differential $\partial u^*/\partial y^*$ is zero and thus the integral on the right-hand side in (2.30) simplifies.
The second term on the left-hand side of equation (2.30) can be simplified by applying the partial integral and using the integrated continuity equation (2.24) in the following way

$$\begin{aligned} \int_0^{h^*} v^* \frac{\partial u^*}{\partial y^*} dy^* &= \left[u^* v^* \right]_0^{h^*} - \int_0^{h^*} u^* \frac{\partial v^*}{\partial y^*} dy^* = \left[u^* v^* \right]_0^{h^*} + \int_0^{h^*} u^* \frac{\partial u^*}{\partial x^*} dy^* \\ &= -U_e^* \int_0^{h^*} \frac{\partial u^*}{\partial x^*} dy^* + \int_0^{h^*} u^* \frac{\partial u^*}{\partial x^*} dy^*. \end{aligned}$$

After inserting the partial integration into equation (2.30) the following simplification is found to be

$$\int_0^{h^*} \frac{\partial}{\partial x^*} \left[u^* \left(U_e^* - u^* \right) \right] dy^* + \frac{dU_e^*}{dx^*} \int_0^{h^*} \left(U_e^* - u^* \right) dy^* = \frac{\tau_w^*}{\rho^*}. \tag{2.32}$$

The final integral boundary-layer equation is derived by introducing the displacement thickness δ_1^* and the momentum thickness δ_2^* into (2.32). Further, $h^* \to \infty$ which does not change the equation as the derivative $\partial u^*/\partial y^*$ vanishes in the outer flow. The integral boundary-layer equation for a two-dimensional and incompressible flow is found

$$\frac{d}{dx^*} \left(U_e^{*2} \delta_2^* \right) + \delta_1^* U_e^* \frac{dU_e^*}{dx^*} = \frac{\tau_w^*}{\rho^*} \quad \text{or} \quad \frac{d\delta_2^*}{dx^*} + \frac{2\delta_2^* + \delta_1^*}{U_e^*} \frac{dU_e^*}{dx^*} = \frac{\tau_w^*}{\rho U_e^{*2}}. \tag{2.33}$$

Further, another important boundary-layer characteristic can be obtain when solving the integral boundary-layer equation (2.33). The local friction coefficient is defined as

$$C_f(x) = \frac{\tau_w^*(x)}{\frac{1}{2} \rho^* U_\infty^{*\,2}} \tag{2.34}$$

with the wall shear stress $\tau_w^*(x)$ given by equation (2.31) and the free-stream velocity U_∞^*.

2.4 Solutions of the Integral Boundary-Layer Equation

The solution of the laminar integral boundary-layer equation can be achieved with less effort than solving the original boundary-layer equation (2.25) because equation (2.33) is an ordinary differential equation. Historically, the integral boundary-layer equation was used to determine boundary-layer characteristics as it could be solved in reasonable time. As computers became more powerful within the last 20 years computations of the boundary layer can nowadays accomplished in a reasonable time by solving the direct boundary-layer equation (2.25). However, the integral boundary-layer equation is still employed for parameter studies and airfoil design programs in turbulent flows and thus methods to solve equation (2.33) are worth to mention here. In the following, the Pohlhausen method is presented and three other methods with are based on Pohlhausen's derivation.
In chapter 3 the second Walz method and Thwaites' method are implemented. The second Walz method is implemented as it is based on the Hartree parameter and referred to as a very exact method for boundary-layer flows [20] whereas Thwaites' method represents a very common method often used in literature [8].

2.4.1 Pohlhausen Method

In order to solve the integral boundary-layer equation (2.33) Pohlhausen [16] assumed a quartic polynomial approach for the dimensionless boundary-layer profiles. This is given by

$$\frac{u^*}{U_e^*} = f(\eta) = a + b\eta + c\eta^2 + d\eta^3 + e\eta^4 \quad \text{with} \quad \eta = \frac{y^*}{\delta^*(x)} \tag{2.35}$$

where δ^* is the wall-normal distance at the edge of the boundary layer where $u \to U_e^*$ corresponding to the local inviscid velocity. The parameter δ^* is commonly known as the boundary-layer thickness. The boundary-layer thickness is increasing in streamwise direction and thus δ^* changes at every streamwise position. The coefficients of equation (2.35) are found by using the following boundary conditions:

$$y^* = 0: \qquad u^*(0) = 0, \qquad \Rightarrow a = 0 \tag{2.36a}$$

$$y^* = \delta^*: \qquad u^*(\delta^*) = U_e^*, \qquad \Rightarrow b + c + d + e = 1 \tag{2.36b}$$

$$y^* = 0: \qquad 0 = U_e^* \frac{dU_e^*}{dx^*} + \nu^* \left.\frac{\partial^2 u^*}{\partial y^{*2}}\right|_{y^*=0}, \qquad \Rightarrow c = -\frac{1}{2}\frac{\delta^{*2}}{\nu^*}\frac{dU_e^*}{dx^*} \tag{2.36c}$$

$$y^* = \delta^*: \qquad \left.\frac{\partial u^*}{\partial y^*}\right|_{y^*=\delta^*} = 0, \qquad \Rightarrow b + 2c + 3d + 4e = 0 \tag{2.36d}$$

$$y^* = \delta^*: \qquad \left.\frac{\partial^2 u^*}{\partial y^{*2}}\right|_{y^*=\delta^*} = 0, \qquad \Rightarrow 2c + 6d + 12e = 0. \tag{2.36e}$$

In the next step the dimensionless pressure gradient parameter Λ is introduced as

$$\Lambda(x^*) = -\frac{\delta^{*2}}{U_e^* \mu^*}\frac{dp^*}{dx^*} = \frac{\delta^{*2}}{\nu^*}\frac{dU_e^*}{dx^*} \tag{2.37}$$

which is also known as the Pohlhausen parameter. The Pohlhausen parameter is a shape function for the velocity polynomial (2.35) that depends on the streamwise position x. By solving the system of equation (2.36) and inserting the parameter Λ it follows that the coefficients are

$$a = 0, \quad b = 2 + \frac{\Lambda}{6}, \quad c = -\frac{\Lambda}{2}, \quad d = -2 + \frac{\Lambda}{2}, \quad e = 1 - \frac{\Lambda}{6}.$$

The quartic polynomial for the velocity profiles is found to be

$$\frac{u^*}{U_e^*} = f(\eta) = 2\eta - 2\eta^3 + \eta^4 + \frac{\Lambda}{6}\eta\left(1-\eta\right)^3 \quad \text{with} \quad \eta = \frac{y^*}{\delta^*}. \tag{2.38}$$

In Figure 2.2 the shape of the velocity profile is shown for different values of the Pohlhausen parameter Λ.

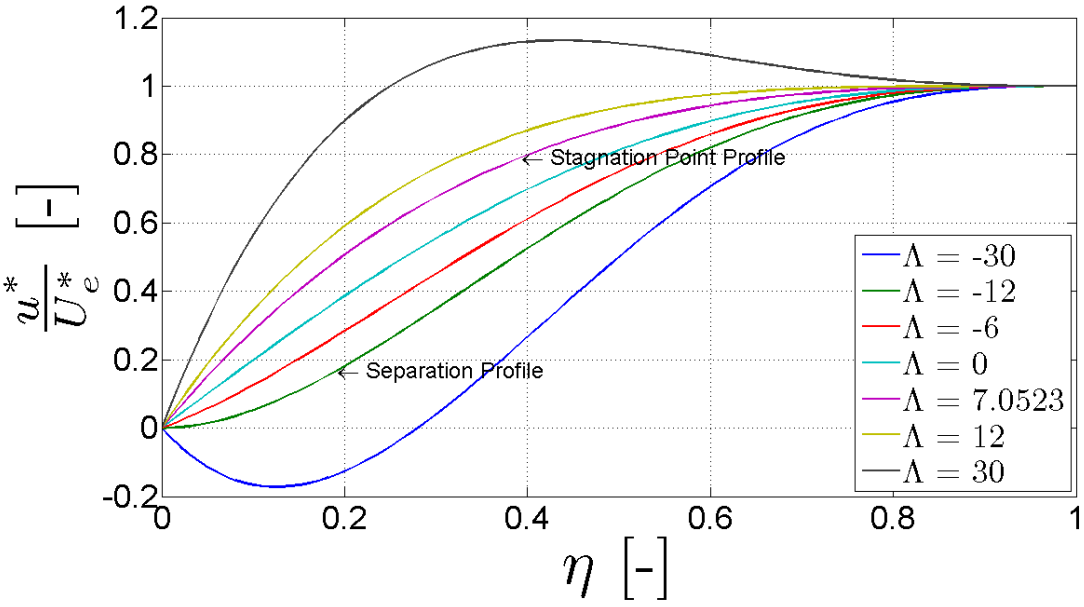


Figure 2.2.: The shape of the velocity profiles as a function of the Pohlhausen parameter Λ. The stagnation point and the separation point profiles are given for $\Lambda = 7.0523$ and $\Lambda = -12$, respectively.

Using the final quartic polynomial the displacement thickness δ_1^*, the momentum thickness δ_2^* and the right-hand side of the integral boundary-layer equation (2.33) can be given as

$$\delta_1^* = \delta^* \int_0^1 \left(1-f\right) d\eta = \delta^* \frac{36-\Lambda}{120} \tag{2.39}$$

$$\delta_2^* = \delta^* \int_0^1 \left[f\left(1-f\right)\right] d\eta = \delta^* \frac{5328 - 48\Lambda - 5\Lambda^2}{45360} \tag{2.40}$$

$$\frac{\tau_w^*}{\rho^* U_e^{*2}} = \frac{\nu^*}{U_e^* \delta^*} f'(0) = \frac{\nu^*}{U_e^* \delta^*}\left(2 + \frac{\Lambda}{6}\right). \tag{2.41}$$

Inserting equation (2.39), (2.40) and (2.41) into the second form of the integral boundary-layer equation (2.33) results into

$$\frac{d}{dx^*}\left(\delta^* \int_0^1 \left[f\left(1-f\right)\right] d\eta\right) + \frac{2\delta^*}{U_e^*}\frac{dU_e^*}{dx^*} \int_0^1 \left[f\left(1-f\right)\right] d\eta + \frac{\delta^*}{U_e^*}\frac{dU_e^*}{dx^*} \int_0^1 \left(1-f\right) d\eta = \frac{\nu^*}{U_e^* \delta^*} f'(0).$$

The integrals do not depend on x^* and by multiplying with δ^*/ν^* it follows

$$\underbrace{\frac{\delta^*}{\nu^*}\frac{dU_e^*}{dx^*}\frac{d\delta^*}{dx^*}\frac{\int_0^1\left[f\left(1-f\right)\right]d\eta}{\frac{dU_e^*}{dx^*}}+\frac{1}{2}\frac{\delta^{*2}}{\nu^*}\frac{d^2U_e^*}{dx^{*2}}\frac{\int_0^1\left[f\left(1-f\right)\right]d\eta}{\frac{dU_e^*}{dx^*}}}_{\frac{d\Lambda}{dx^*}\frac{\int_0^1[f(1-f)]d\eta}{\frac{dU_e^*}{dx^*}}}-$$

$$-\frac{1}{2}\frac{\delta^{*2}}{\nu^*}\frac{d^2U_e^*}{dx^{*2}}\frac{\int_0^1\left[f\left(1-f\right)\right]d\eta}{\frac{dU_e^*}{dx^*}}+\frac{1}{U_e^*}\frac{\delta^{*2}}{\nu^*}\frac{dU_e^*}{dx^*}\left[2\int_0^1\left[f\left(1-f\right)\right]d\eta+\int_0^1\left(1-f\right)d\eta\right]=\frac{f'(0)}{U_e^*}.$$

Note that the second and third term are just an addition and a subtraction. This is done to simplify the first two terms to the term shown under the bracket. By multiplying the equation above with $\frac{dU_e^*}{dx^*}$, dividing by $\int_0^1\left[f\left(1-f\right)\right]d\eta$ and using the definition (2.37) the simplified equation reads

$$\frac{d\Lambda}{dx^*}=\frac{1}{U_e^*}\frac{dU_e^*}{dx^*}\underbrace{\left[\frac{f'(0)}{\int_0^1\left[f\left(1-f\right)\right]d\eta}-\Lambda\left(2+\frac{\int_0^1\left(1-f\right)d\eta}{\int_0^1\left[f\left(1-f\right)\right]d\eta}\right)\right]}_{g(\Lambda)}+\underbrace{\frac{\Lambda}{2}}_{h(\Lambda)}\frac{\frac{d^2U_e^*}{dx^{*2}}}{\frac{dU_e^*}{dx^*}} \tag{2.42}$$

or in short form

$$\frac{d\Lambda}{dx^*}=\frac{1}{U_e^*}\frac{dU_e^*}{dx^*}g(\Lambda)+\frac{\frac{d^2U_e^*}{dx^{*2}}}{\frac{dU_e^*}{dx^*}}h(\Lambda) \tag{2.43}$$

with

$$g(\Lambda)=-\frac{378}{5}-2\Lambda+\frac{48384\left(\Lambda-51\right)}{5\left[-5328+\Lambda\left(48+5\Lambda\right)\right]} \tag{2.44a}$$

$$h(\Lambda)=\frac{\Lambda}{2}. \tag{2.44b}$$

The functions for $g(\Lambda)$ and $h(\Lambda)$ are found by inserting the polynomial (2.35) which will result in the two functions g and h that only depend on $\Lambda(x^*)$. In order to find the physical values for $\Lambda(x^*)$ and $\delta^*(x^*)$ equation (2.42) has to be integrated which can be done numerically or with graphic solution methods like the isocline method [19]. Thereby, the Pohlhausen parameter Λ is found as a function of x^*.

The Pohlhausen method is only valid for steady boundary-layer flows. The shape paramter $\Lambda \leq 12$ has to be ensured as otherwise the maximum velocity inside the boundary layer becomes larger than the edge velocity which is not possible. The other limit is reached for $\Lambda = -12$ which characterizes the separation profile. For values lower than $\Lambda = -12$ the boundary-layer equations on which the assumptions of Pohlhausen are based on are not valid. Those two limits give the region of application for the Pohlhausen method. To illustrate those prerequisites, profiles within and outside the range of application are shown in Figure 2.2.

The integration of (2.43) has to start at the stagnation point ($x^* = 0$) and as the edge velocity is zero at that point a singularity occurs. However, the function $g(\Lambda)$ has also a root that results in an initial value for Λ which is $\Lambda_0 = 7.0523$. Thereby, the singularity problem is resolved as the derivatives of the edge velocity do not become zero.

After obtaining $\Lambda(x^*)$ and $\delta^*(x^*)$ from the integration of (2.42) the characteristic boundary-layer values for the displacement and momentum thickness are found by inserting into equations (2.39) and (2.40).

2.4.2 First Walz Method

In contrast to Pohlhausen's method which depends on the somewhat arbitrarily chosen parameter δ^*, the more accurate and exactly defined momentum thickness δ_2^* can be used to define a different shape parameter

$$\begin{aligned} \lambda &= -\frac{\delta_2^{*2}}{U_e^* \mu^*} \frac{dp^*}{dx^*} = \underbrace{\frac{\delta_2^{*2}}{\nu^*}}_{Z} \frac{dU_e^*}{dx^*} \\ \lambda &= Z^* \frac{dU_e^*}{dx^*} \end{aligned} \tag{2.45}$$

that also depends on the results from the potential flow solution. The term below the underbraces is used to define the parameter Z^* for which the equation is later on solved for.
By multiplying the integral boundary-layer equation (2.33) with $U_e^* \delta_2^* / \nu$ it follows

$$\frac{U_e^* \delta_2^*}{\nu^*} \frac{d\delta_2^*}{dx^*} + \frac{\delta_2^{*2}}{\nu^*} \left(2 + \frac{\delta_1^*}{\delta_2^*} \right) \frac{dU_e^*}{dx^*} = \frac{\delta_2^*}{U_e^*} \left. \frac{\partial u^*}{\partial y^*} \right|_{y^*=0}$$

which can be simplified with the definition of Z^* from equation (2.45) to

$$\frac{dZ^*}{dx^*} = \frac{F(\lambda)}{U_e^*}. \tag{2.46}$$

The function $F(\lambda)$ is found by using the Pohlhausen parameter (2.37) and converting to the shape parameter λ which is now based on the momentum thickness

$$\lambda = \frac{\delta_2^{*2}}{\delta^{*2}} \Lambda = \left(\frac{5328 - 48\Lambda - 5\Lambda^2}{45360} \right)^2 \Lambda. \tag{2.47}$$

The function $F(\lambda)$ consists of

$$F(\lambda) = 2 \left[\frac{\delta_2^*}{U_e^*} \left. \frac{\partial u^*}{\partial y^*} \right|_{y^*=0} - \lambda \left(2 + \frac{\delta_1^*}{\delta_2^*} \right) \right] = 2G(\lambda) - 4\lambda - 2\lambda H(\lambda) \tag{2.48}$$

with the subfunctions

$$G(\lambda) = \frac{\delta_2^*}{U_e^*} \left. \frac{\partial u^*}{\partial y^*} \right|_{y^*=0} = \left(2 + \frac{\Lambda}{6} \right) \frac{5328 - 48\Lambda - 5\Lambda^2}{45360} \tag{2.49a}$$

$$H(\lambda) = \frac{\delta_1^*}{\delta_2^*} = \frac{\frac{36-\Lambda}{120}}{\frac{5328-48\Lambda-5\Lambda^2}{45360}}. \tag{2.49b}$$

Equation (2.46) is a nonlinear, first order, ordinary differential equation. It has to be solved in a similar way as the Pohlhausen equation (2.42). The initial value, which is needed to start the computationm $\lambda_0 = 0.0770$ is obtained by employing Λ_0 from the Pohlhausen method in equation (2.47).
A major advantage of using the momentum thickness instead of the boundary-layer thickness results owes the fact that the second derivative of the edge velocity does not appear anymore in (2.46). Only

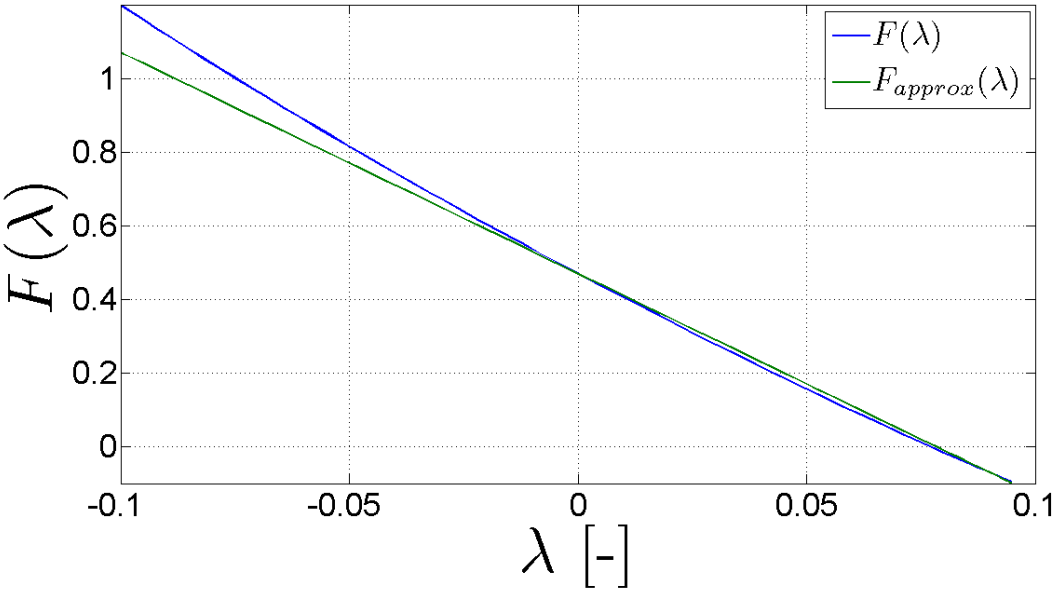


Figure 2.3.: The universal function $F(\lambda)$ needed to integrate the differential equation (2.46) and the approximation to find the approximate solution given in equation (2.52).

the first derivative is used to compute $\lambda(x^*)$ with equation (2.45) after having found $Z^*(x^*)$.
As a matter of fact the function $F(\lambda)$ is almost linear and in order to simplify the solution of the nonlinear differential equation (2.46) the function is approximated by [19]

$$F(\lambda) \approx a - b\lambda \qquad \text{with} \quad a = 0.47 \quad \text{and} \quad b = 6. \tag{2.50}$$

The difference between $F(\lambda)$ and the approximation given in equation (2.50) is depicted in Figure 2.3. The simplified differential equation reads

$$U_e^* \frac{dZ^*}{dx^*} = a - b\lambda$$

or

$$\frac{d}{dx^*}\left(U_e^* Z^*\right) = a - (b-1)U_e^* Z^* \frac{1}{U_e^*}\frac{dU_e^*}{dx^*}. \tag{2.51}$$

The neat fact by using that linearization is the solution in closed form which can be used to directly compute the results for the boundary-layer characteristics. The integration of (2.51) yields

$$U_e^* Z^* = \frac{a}{U_e^{*b-1}} \int_0^{x^*} U_e^{*b-1} dx^* \tag{2.52}$$

which gives Z^* and in the following λ, Λ and all the characteristics.
The linearized function $F(\lambda)$ is very close to the nonlinear function in regions with favorable pressure gradient ($\lambda > 0$) as shown in Figure 2.3. The errors in those regions are thus very small. However, in case of adverse pressure gradients the linearization becomes worse. The separation profile is found for $\lambda = -0.1567$ which is far off the exact function $F(\lambda)$ as illustrated in Figure 2.3. Therefore, the calculated characteristics increase in error for streamwise positions approaching the separation point.

2.4.3 Second Walz Method

In contrast to the Pohlhausen method and the first Walz method where the shape parameter is based on the pressure term when solving the boundary-layer equation at the wall, the second Walz method uses the viscous term as the shape parameter. Further, the profile family is based on the Hartree profiles [26], which are results of the Falkner-Skan equation that is given in section 2.5.1. By multiplying the compatibility condition at the wall it follows

$$\underbrace{-\frac{\delta_2^{*2}}{U_e^*}\left(\frac{\partial^2 u^*}{\partial y^{*2}}\right)\bigg|_{y^*=0}}_{\Gamma} = \underbrace{\frac{\delta_2^{*2}U_e^*}{\nu^*}}_{Z^*}\frac{1}{U_e^*}\frac{dU_e^*}{dx^*}.$$

The integral boundary-layer equation (2.33) is multiplied with $(2\delta_2^* U_e^*)/\nu$ and reads

$$\frac{d}{dx^*}\left(\frac{U_e^*\delta_2^{*2}}{\nu}\right) - \frac{\delta_2^{*2}}{\nu}\frac{dU_e^*}{dx^*} + 2\frac{\delta_2^{*2}}{\nu}\frac{dU_e^*}{dx^*}\left(2+\frac{\delta_1^*}{\delta_2^*}\right) = 2\frac{\tau_w^*\delta_2^*}{\rho^*\nu^*U_e^*}.$$

The coupled system of equation then becomes in short form

$$\frac{dZ^*}{dx^*} = F(\Gamma) \tag{2.53}$$

$$\frac{Z^*}{U_e^*}\frac{dU_e^*}{dx^*} = \Gamma \tag{2.54}$$

with

$$F(\Gamma) = 2\frac{\tau_w^*\delta_2^*}{\rho^*\nu^*U_e^*} - \left(3+2\frac{\delta_1^*}{\delta_2^*}\right)\Gamma. \tag{2.55}$$

In order to obtain the characteristic values of the boundary layer the coupled system of the two equations (2.53) and (2.54) for the unknowns $Z^*(x^*)$ and $\Gamma(x^*)$ has to be solved which is again a nonlinear first order differential equation. The principal advantage of that derivation is the exact solution for all wedge flows [20].
Like in the first Walz method the function $F(\Gamma)$ is almost linear and thus will be approximated with the linear function

$$F(\Gamma) = a - b\Gamma. \tag{2.56}$$

In Figure 2.4 the approximated functions are plotted against Γ. The two approximation functions $F_{approx1}(\Gamma)$ and $F_{approx2}(\Gamma)$ are based on two different values given in the books of Walz [26] and Schlichting [20]. For the second Walz method the constants are assigned in a way that the solution is exact for a plate flow ($\Gamma = 0$), a stagnation-point flow ($\Gamma = 0.0855$) and a separation flow ($\Gamma = -0.0681$). The parameters a and b are thus different for the two different regions and read for $F_{approx1}(\Gamma)$ [26]

$$\begin{aligned} \Gamma > 0: &\quad a = 0.441, \quad b = 4.165 \\ \Gamma < 0: &\quad a = 0.441, \quad b = 5.165 \end{aligned} \tag{2.57}$$

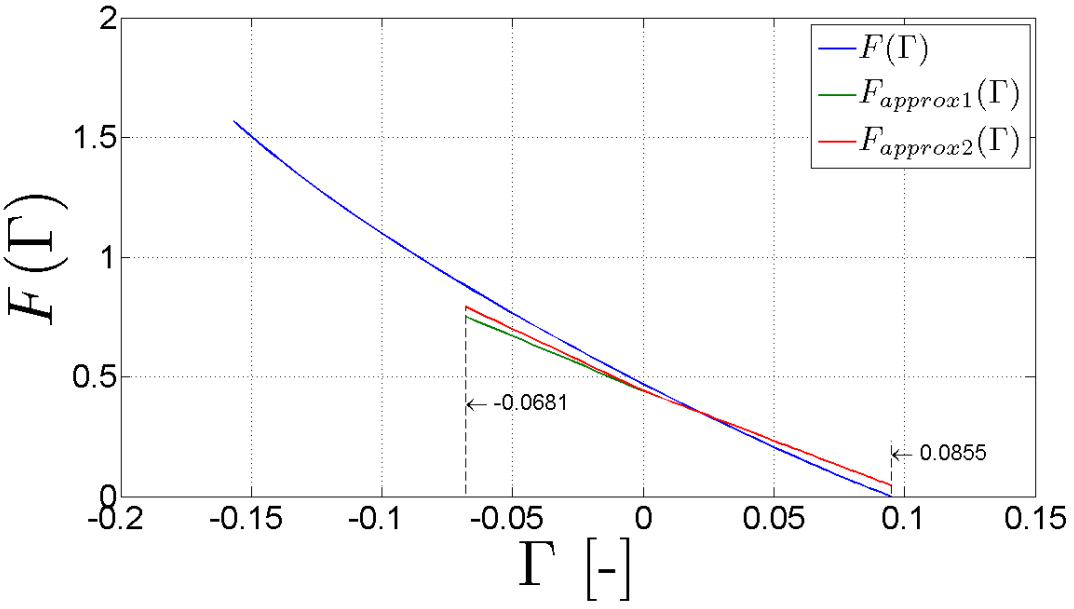


Figure 2.4.: The universal function $F(\Gamma)$ needed to integrate the differential equation (2.53) and the approximations to find the approximate solution given in equation (2.52) with the two different values based on Walz [26] and Schlichting [20].

and for $F_{approx2}(\Gamma)$ [20]

$$\begin{aligned} \Gamma > 0: \quad & a = 0.441, \quad b = 4.165 \\ \Gamma < 0: \quad & a = 0.441, \quad b = 4.579. \end{aligned} \tag{2.58}$$

Inserting the linearization into equations (2.53) and (2.54) gives the ordinary differential equation

$$\frac{dZ^*}{dx^*} + \frac{b}{U_e^*}\frac{dU_e^*}{dx^*}Z^* = a \tag{2.59}$$

which can be solved analytically such that

$$Z^*(x^*) = \frac{a}{\left[U_e^*(x^*)\right]^b}\int_0^{x^*}\left[U_e^*(x^*)\right]^b dx^*. \tag{2.60}$$

2.4.4 Thwaites Method

Like in the first Walz method, Thwaites uses for his method the momentum thickness to define the shape parameter λ which is already given in equation (2.45). Further, another dimensionless parameter G is introduced as

$$G(\lambda) = \frac{\delta_2^*}{U_e^*}\frac{du^*}{dy^*}\bigg|_{y^*=0} \tag{2.61}$$

which is already given in the first Walz method and substituted in the integral boundary-layer equation (2.33). It follows that

$$\frac{U_e^* \delta_2^*}{\nu^*} \frac{d\delta_2^*}{dx^*} + \left(2 + \frac{\delta_1^*}{\delta_2^*}\right) \lambda = G(\lambda)$$

and a final rearrangement gives

$$\frac{U_e^*}{\nu^*} \frac{d\delta_2^{*2}}{dx^*} = 2 \left[G(\lambda) - \left(2 + \frac{\delta_1^*}{\delta_2^*}\right) \lambda \right] \tag{2.62}$$

In equation (2.62), Thwaites assumes that the right-hand side is a universal function that simply depends on λ in the following way

$$F(\lambda) \approx 0.45 - 6\lambda \approx 2 \left[G(\lambda) - \left(2 + \frac{\delta_1^*}{\delta_2^*}\right) \lambda \right]. \tag{2.63}$$

Another substitution of equation (2.63) into equation (2.62) and a multiplication with U_e^{*5} gives the simple equation

$$\frac{d\left(\delta_2^{*2} U_e^{*6}\right)}{dx^*} = 0.45 \nu^* U_e^{*5} \tag{2.64}$$

that can be solved to get δ_2^* for the known function of U_e^* such that

$$\delta_2^{*2}(x^*) = \delta_2^{*2}(0) \left(\frac{U_e^*(0)}{U_e^*(x^*)} \right)^6 + \frac{0.45\nu^*}{U_e^{*6}(x^*)} \int_0^{x^*} U_e^{*5}(x^*) dx^*. \tag{2.65}$$

For a stagnation-point flow, the edge velocity at the stagnation point is zero and according to Cebeci [8] the momentum thickness for the stagnation point is given by

$$\delta_2^{*2}(0) = \frac{0.075\nu^*}{\left.\frac{dU_e^*}{dx^*}\right|_{x^*=0}}. \tag{2.66}$$

For dimensionless values equations (2.65) and (2.66) read

$$\delta_2^2(x) = \delta_2^2(0) \left(\frac{U_e(0)}{U_e(x)} \right)^6 + \frac{0.45}{Re U_e{}^6(x)} \int_0^x U_e{}^5(x) dx \tag{2.67}$$

$$\delta_2^2(0) = \frac{0.075}{Re \left.\frac{dU_e}{dx}\right|_{x=0}}. \tag{2.68}$$

After having obtained $\delta_2(x)$ the pressure gradient parameter λ can be found. In a next step the following relations are used to get the dimensionless skin friction parameter l and the shape factor $H_{12} = \frac{\delta_1}{\delta_2}$

$$G(\lambda) \approx \begin{cases} 0.22 + 1.402\lambda + \frac{0.018\lambda}{0.107+\lambda}, & \text{for } -0.1 \le \lambda \le 0 \\ 0.22 + 1.57\lambda - 1.8\lambda^2, & \text{for } 0 \le \lambda \le 0.1 \end{cases} \tag{2.69}$$

$$H_{12}(\lambda) \approx \begin{cases} 2.088 + \frac{0.0731}{0.14+\lambda}, & \text{for } -0.1 \le \lambda \le 0 \\ 2.61 - 3.75\lambda + 5.24\lambda^2, & \text{for } 0 \le \lambda \le 0.1 \end{cases} \tag{2.70}$$

2.5 Transformations of the Boundary-Layer Equation

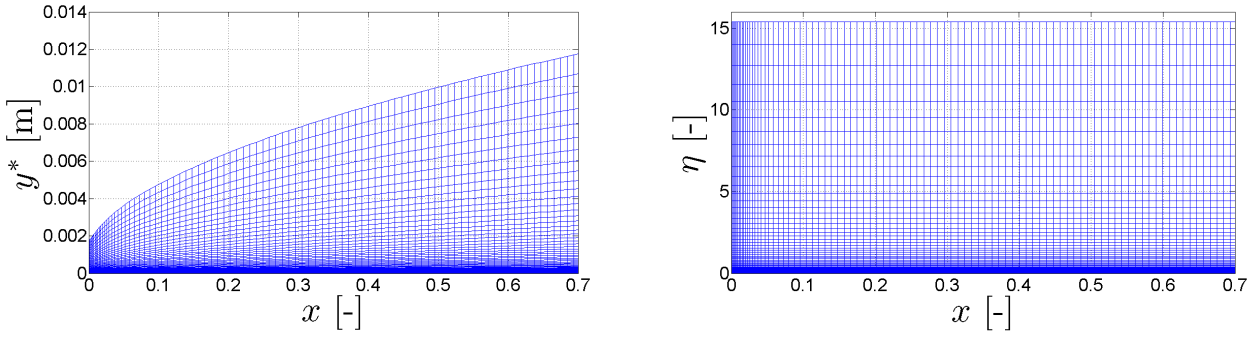
In contrast to the methods shown for the integral boundary-layer equation the direct boundary-layer equation derived in section 2.2 is a partial differential equation that is analytically impossible to solve. In order to simplify the numerical solution of the direct boundary-layer equation some transformations are introduced. In the following the two most common boundary-layer transformations are presented.

2.5.1 The Falkner-Skan Transformation

The boundary-layer equation (2.25) is a simplification of the Navier-Stokes equation for flows close to the wall at high Reynolds numbers. However, equation (2.25) is still a partial differential equation. By introducing a suitable transformation the solution can be simplified. An often used scaling for the wall-normal direction is given by

$$\eta = y^* \sqrt{\frac{U_e^*(x^*)}{\nu^* x^*}} = y \sqrt{Re \frac{U_e(x)}{x}}. \tag{2.71}$$

Thereby, the wall-normal quantity y is stretched in order to have a similar order of magnitude as for the streamwise direction. The effect of that scaling is shown in Figure 2.5. The maximum value of y is increasing in streamwise direction as depicted in Figure 2.5a due to the boundary-layer thickness increase. In order to fulfill the boundary condition in the free-stream (2.27) the computation domain in wall-normal direction must increase. In contrast to that the boundary scaling in wall-normal direction (2.71) avoids the rise of the computational domain as shown in Figure 2.5b.



(a) Physical Domain **(b)** Dimensionless Domain

Figure 2.5.: Comparison between physical and dimensionless computation domain based on the numerical grid.

In a next step the stream function

$$\psi^*(x,y) = \sqrt{\nu^* U_e^*(x^*) x^*} f(x^*,\eta) = U_{ref}^* L_{ref}^* \sqrt{\frac{U_e(x)x}{Re}} f(x,\eta) \tag{2.72}$$

is introduced which fulfills the continuity equation (2.24). The function f can be interpreted as a dimensionless stream function that depends on x and η. The length scale L_{ref}^* and velocity scale U_{ref}^* can be chosen arbitrarily, common values are the chord length L_c^* and the free-stream velocity U_∞^*. Those

values are also used to define the global Reynolds number. In order to insert the transformation (2.71) and the stream function (2.72) into the boundary-layer equation (2.25) several derivatives have to be taken. Those derivatives read

$$\frac{\partial \eta}{\partial x^*} = \frac{1}{2}\eta\left(\frac{\partial U_e^*/\partial x^*}{U_e^*} - \frac{1}{x^*}\right) = \frac{1}{L_{ref}^*}\frac{1}{2}\eta\left(\frac{\partial U_e/\partial x}{U_e} - \frac{1}{x}\right) \tag{2.73a}$$

$$\frac{\partial \eta}{\partial y^*} = \sqrt{\frac{U_e^*}{\nu^* x^*}} = \frac{1}{L_{ref}^*}\sqrt{Re\frac{U_e}{x}} \tag{2.73b}$$

$$\frac{\partial \psi^*}{\partial x^*} = \sqrt{\nu^* U_e^* x^*}\left[f_{x^*} + \frac{1}{2}f\left(\frac{\partial U_e^*/\partial x^*}{U_e^*} + \frac{1}{x^*}\right)\right] = U_{ref}^*\sqrt{\frac{U_e x}{Re}}\left[f_x + \frac{1}{2}f\left(\frac{\partial U_e/\partial x}{U_e} + \frac{1}{x}\right)\right] \tag{2.73c}$$

$$\frac{\partial \psi^*}{\partial \eta} = \sqrt{\nu^* U_e^* x^*} f_\eta = L_{ref}^* U_{ref}^* \sqrt{\frac{U_e x}{Re}} f_\eta \tag{2.73d}$$

in dimensional and dimensionless form. Note that the physical values are included in the dimensionless forms due to the reference length and velocity. The subscript below the function f means to take a derivative with respect to the coordinate, e.g. f_x means to derive f with respect to x and f_η means to take the derivative of the function f with respect to η. Consequently, two or more subscripts imply derivatives with respect to two or more subscripts.
With those simplifications the continuity equation (2.24) is fulfilled and each term of the boundary-layer equation can be found by using the simplifications (2.73). Those terms are shown below:

$$u^* = \frac{\partial \psi^*}{\partial y^*} = \frac{\partial \psi^*}{\partial \eta}\frac{\partial \eta}{\partial y^*} = U_e^* f_\eta = U_{ref}^* U_e f_\eta \tag{2.74a}$$

$$\begin{aligned} v^* &= -\frac{\partial \psi^*}{\partial x^*} = -\frac{\partial \psi^*}{\partial x^*} - \frac{\partial \psi^*}{\partial \eta}\frac{\partial \eta}{\partial x^*} \\ v^* &= -\sqrt{\nu^* U_e^* x^*}\left[\frac{1}{2}f\left(\frac{\partial U_e^*/\partial x^*}{U_e^*} + \frac{1}{x^*}\right) + f_{x^*} + \frac{1}{2}f_\eta\eta\left(\frac{\partial U_e^*/\partial x^*}{U_e^*} - \frac{1}{x^*}\right)\right] \\ v^* &= -U_{ref}^*\sqrt{\frac{U_e x}{Re}}\left[\frac{1}{2}f\left(\frac{\partial U_e/\partial x}{U_e} + \frac{1}{x}\right) + f_x + \frac{1}{2}f_\eta\eta\left(\frac{\partial U_e/\partial x}{U_e} - \frac{1}{x}\right)\right] \end{aligned} \tag{2.74b}$$

$$\begin{aligned} \frac{\partial u^*}{\partial x^*} &= \frac{\partial u^*}{\partial x^*} + \frac{\partial u^*}{\partial \eta}\frac{\partial \eta}{\partial x^*} \\ \frac{\partial u^*}{\partial x^*} &= \frac{\partial U_e^*}{\partial x^*} f_\eta + U_e^* f_{x^*\eta} + \frac{1}{2}f_{\eta\eta}\eta U_e^*\left(\frac{\partial U_e^*/\partial x^*}{U_e^*} - \frac{1}{x^*}\right) \\ \frac{\partial u^*}{\partial x^*} &= \frac{U_{ref}^*}{L_{ref}^*}\left[\frac{\partial U_e}{\partial x} f_\eta + U_e f_{x\eta} + \frac{1}{2}f_{\eta\eta}\eta U_e\left(\frac{\partial U_e/\partial x}{U_e} - \frac{1}{x}\right)\right] \end{aligned} \tag{2.74c}$$

$$\begin{aligned} \frac{\partial u^*}{\partial y^*} &= \frac{\partial u^*}{\partial \eta}\frac{\partial \eta}{\partial y^*} \\ \frac{\partial u^*}{\partial y^*} &= U_e^*\sqrt{\frac{U_e^*}{\nu^* x^*}} f_{\eta\eta} = \frac{U_{ref}^*}{L_{ref}^*} U_e \sqrt{Re\frac{U_e}{x}} f_{\eta\eta} \end{aligned} \tag{2.74d}$$

$$\begin{aligned} \frac{\partial^2 u^*}{\partial y^{*2}} &= \frac{\partial^2 u^*}{\partial \eta^2}\left(\frac{\partial \eta}{\partial y^*}\right)^2 \\ \frac{\partial^2 u^*}{\partial y^{*2}} &= \frac{U_e^{*2}}{\nu^* x^*} f_{\eta\eta\eta} = \frac{U_{ref}^*}{{L_{ref}^*}^2} Re\frac{U_e^2}{x} f_{\eta\eta\eta}. \end{aligned} \tag{2.74e}$$

Again, the terms are given in dimensional values as well as dimensionless values with includded reference scales. By applying those derivatives to the boundary-layer equation (2.25) it reads

$$f_{\eta\eta\eta} + \frac{m+1}{2} f f_{\eta\eta} + m\left(1 - f_\eta^2\right) = x\left(f_\eta f_{x\eta} - f_x f_{\eta\eta}\right) \tag{2.75}$$

where m is a dimensionless pressure gradient defined by

$$m = \frac{x^*}{U_e^*}\frac{\partial U_e^*}{\partial x^*} = \frac{x}{U_e}\frac{\partial U_e}{\partial x}. \tag{2.76}$$

Note that the left- and right-hand side of equation (2.75) are given in dimensionless terms.
The boundary conditions for equation (2.75) are directly derived from the boundary conditions of the original boundary-layer equation (2.27). In order to fulfill the no-slip condition at the wall $u(x,0) = v(x,0) = 0$ and the potential flow condition at the boundary-layer edge $u(x, y \to \infty) = U_e(x)$ the following boundary conditions

$$\begin{aligned} \eta = 0: &\qquad f = 0 \quad \text{and} \quad f_\eta = 0 \\ \eta \to \infty: &\qquad f_\eta = 1 \end{aligned} \tag{2.77}$$

must be satisfied.
The general form of such a boundary-layer transformation (2.75) can be used to solve a boundary-layer of an arbitrarily shaped body, e.g. an airfoil. However, if the dimensionless pressure gradient (2.76) is constant the right-hand side of equation (2.75) cancels out and reduces to

$$f_{\eta\eta\eta} + \frac{m+1}{2} f f_{\eta\eta} + m\left(1 - f_\eta^2\right) = 0 \tag{2.78}$$

and the solution is the so-called self-similar solution because it does not depend of the streamwise component x any more. That equation was derived for the first time by Falkner and Skan and is therefore called *Falkner-Skan Equation*. Consequently, the transformation (2.71) with the stream function (2.72) is called *Falkner-Skan transformation* in the following work. Furthermore, if the right-hand side cancels out the system of partial differential equations reduces to a system of ordinary differential equations which simplifies the solution dramatically.
Such a solution where the pressure gradient is constant is valid for wedge flows. The stream function for a wedge flow is given in [22] as

$$\psi_{\text{wedge}}(r,\varphi) = \frac{a}{n} r^n \sin\left(n\varphi - \pi(n-1)\right) \tag{2.79}$$

where r and φ are the independent coordinates in a cylindrical coordinate system and a is a constant. A wedge flow is described by the stream function above for values of $1 \leq n \leq 2$ where $n = 1$ results in a flat plate flow and $n = 2$ yields the stagnation flow. The absolute velocity for such a flow is [22]

$$|\vec{u}| = |a|\, r^{n-1} \tag{2.80}$$

where a is a constant. The stream function at the wall $\psi_{\text{wedge}} = 0$ results in a physical wedge angle as shown in Figure 2.6. By defining $n - 1 = m$, the wedge angle

$$\beta = \pi\frac{n-1}{n} = \pi\frac{m}{m+1} \tag{2.81}$$

is given. The constant a can be defined arbitrarily and thus it is chosen to be unity in the following. Further, as the stream function at the wall is chosen the coordinate r resembles the coordinate x in the Cartesian coordinate system. With those assumptions the velocity distribution for a wedge flow is found to be

$$U_e(x) = x^m \quad \text{or} \quad U_e^*(x^*) = U_{ref}^* \left(\frac{x^*}{L_{ref}^*} \right)^m . \tag{2.82}$$

The velocity U_{ref}^* is the edge velocity evaluated at L_{ref}^*. Note that m is the dimensionless pressure gradient derived above in equation (2.76).

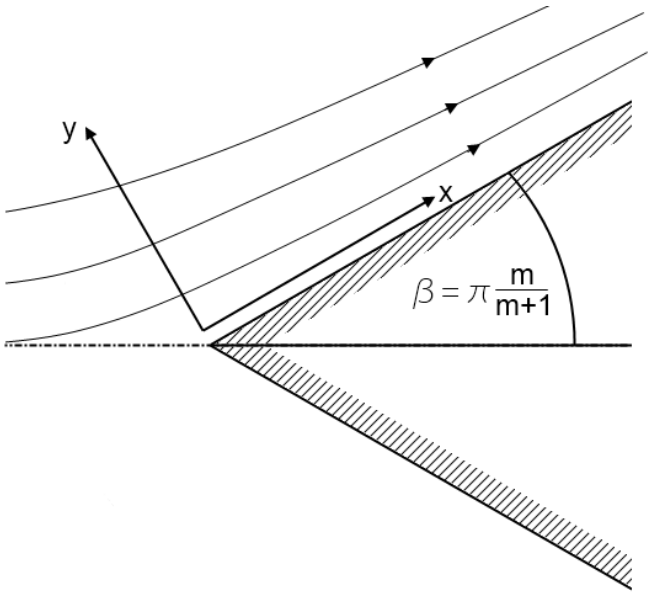


Figure 2.6.: The dimensionless pressure gradient m can be used to define the wedge angle β.

Sometimes a slightly different transformation is used that is referred to Falkner-Skan transformation as well in literature [20]. That transformation reads

$$\eta_\beta = \sqrt{\frac{m+1}{2}}\eta = y^* \sqrt{\frac{m+1}{2}\frac{U_e^*(x^*)}{\nu^* x^*}} = y\sqrt{\frac{m+1}{2}Re\frac{U_e(x)}{x}} \tag{2.83}$$

with the stream function

$$\psi_\beta^*(x,y) = \sqrt{\frac{2}{m+1}\nu^* U_e^*(x^*)x^*} f(x^*,\eta_\beta) = U_{ref}^* L_{ref}^* \sqrt{\frac{2}{m+1}\frac{U_e(x)x}{Re}} f(x,\eta_\beta). \tag{2.84}$$

In order to distinguish the transformation from the original Falkner-Skan transformation (2.71) the subscript β is added. Note that in this case no differentiation with respect to β is meant. If that transformation is inserted into the boundary-layer equation (2.25) the transformed boundary-layer equation is found to be

$$f_{\eta_\beta\eta_\beta\eta_\beta} + f f_{\eta_\beta\eta_\beta} + \beta_H \left(1 - f_{\eta_\beta}^2\right) = 2x \left(f_{\eta_\beta} f_{x\eta_\beta} - f_x f_{\eta_\beta\eta_\beta} \right) \tag{2.85}$$

with the Hartree parameter

$$\beta_H = \frac{2m}{m+1}. \tag{2.86}$$

The boundary conditions do not change and are given in equation (2.77). Again, for similarity solutions like a wedge flow the right-hand side of equation (2.85) cancels out and thus read

$$f_{\eta_\beta\eta_\beta\eta_\beta} + f f_{\eta_\beta\eta_\beta} + \beta_H \left(1 - f_{\eta_\beta}^2\right) = 0. \tag{2.87}$$

As already mentioned above the solution simplifies as equation (2.87) is an ordinary, nonlinear third-order differential equation. Note that the streamwise and wall-normal velocity components for similarity flows are found to be

$$u^* = U_{ref}^* U_e f_\eta \tag{2.88a}$$

$$v^* = -U_{ref}^* \sqrt{\frac{m+1}{2}\frac{x^{m-1}}{Re}} \left(f + \frac{m-1}{m+1}\eta f_\eta\right). \tag{2.88b}$$

Boundary-Layer Characteristics

By solving the boundary-layer equations with the Falkner-Skan transformation, the integral values δ_1 for the displacement thickness (2.28) and δ_2 for the momentum thickness (2.29) can be derived as

$$\delta_1^* = \sqrt{\frac{\nu^* x^*}{U_e^*}}\eta_0(x) = L_{ref}^* \sqrt{\frac{x}{ReU_e}}\eta_0(x) \qquad \text{with} \quad \eta_0(x) = \int_0^\infty \left(1 - f_\eta\right) d\eta = \lim_{\eta\to\infty}\left\{\eta - f(x,\eta)\right\} \tag{2.89}$$

$$\delta_2^* = \sqrt{\frac{\nu^* x^*}{U_e^*}} \int_0^\infty f_\eta \left(1 - f_\eta\right) d\eta = L_{ref}^* \sqrt{\frac{x}{ReU_e}} \int_0^\infty f_\eta \left(1 - f_\eta\right) d\eta. \tag{2.90}$$

Further, the friction coefficient C_f which is characterized by the local shear stress at the wall (2.31) is found with the partial derivative $\frac{\partial u^*}{\partial y^*}$ that is taken from equation (2.74d). By evaluating the derivative at the wall ($\eta = 0$) the local friction coefficient defined by equation (2.34) is

$$C_f = 2\frac{U_e^*}{U_\infty^{*\,2}}\sqrt{\frac{\nu^* U_e^*}{x^*}} f_{\eta\eta}(x,0) = 2U_e\sqrt{\frac{U_e}{Rex}} f_{\eta\eta}(x,0) \tag{2.91}$$

where U_∞^* is the free stream velocity at infinity.

2.5.2 Görtler Transformation

As already mentioned earlier, the boundary-layer equation (2.25) is a partial differential equation which is difficult to solve. In order to simplify the solution, Görtler [13] came up with a suitable transformation that simplifies the boundary-layer equation (2.25) and resolves into an ordinary differential equation for similarity solutions. In contrast to the Falkner-Skan solution, Görtler's transformation considers the growth of the boundary-layer thickness in the wall-normal parameter η which is especially of great advantage for numerical solutions [20].

Whereas the Falkner-Skan transformation (2.71) is applied only in wall-normal direction, Görtler transformed also the streamwise direction. The transformations are

$$\xi = \frac{1}{\nu^*} \int_0^{x^*} U_e^*(x^*)\, dx^* = Re \int_0^{x} U_e(x)\, dx \tag{2.92}$$

$$\eta = y^* \frac{U_e^*(x^*)}{2\nu^* \int_0^{x^*} U_e^*(x^*)\, dx^*} = y^* \frac{U_e^*(x^*)}{\nu^* \sqrt{2\xi}} = y \frac{U_e(x) Re}{\sqrt{2\xi}} \tag{2.93}$$

which are given for dimensional and dimensionless input variables. In a next step a stream function ψ is introduced that fulfills the continuity equation (2.24)

$$\psi^*(x, y) = \nu^* \sqrt{2\xi} F(\xi, \eta) = \frac{U_{ref}^* L_{ref}^*}{Re} \sqrt{2\xi} F(\xi, \eta) \tag{2.94}$$

where F is a dimensionless stream function that depends on the transformation variables ξ and η. In order to insert the transformation into the boundary-layer equation (2.25) the derivatives of ξ and η with respect to x^* and y^* are taken and the derivatives of the stream function ψ^* with respect to ξ and η need to be computed. The results are

$$\frac{\partial \xi}{\partial x^*} = \frac{1}{L_{ref}^*} Re U_e \tag{2.95a}$$

$$\frac{\partial \xi}{\partial y^*} = 0 \tag{2.95b}$$

$$\frac{\partial \eta}{\partial x^*} = \frac{1}{L_{ref}^*} \eta \left(\frac{\partial U_e / \partial x}{U_e} - \frac{Re U_e}{2\xi} \right) \tag{2.95c}$$

$$\frac{\partial \eta}{\partial y^*} = \frac{1}{L_{ref}^*} \frac{U_e Re}{\sqrt{2\xi}} \tag{2.95d}$$

$$\frac{\partial \psi^*}{\partial \xi} = \frac{U_{ref}^* L_{ref}^*}{Re} \sqrt{2\xi} \left(F_\xi + \frac{1}{2\xi} F \right) \tag{2.95e}$$

$$\frac{\partial \psi^*}{\partial \eta} = \frac{U_{ref}^* L_{ref}^*}{Re} \sqrt{2\xi} F_\eta \tag{2.95f}$$

The same notation as in section 2.5.1 is applied where subscripts stand for derivatives with respect to the coordinate, e.g. F_ξ implies a derivative of F with respect to ξ. The derivatives (2.95) are used to

calculate each term of the boundary-layer equation (2.25) in terms of Görtler's transformation variables ξ, η and F. The results are shown below:

$$u^* = \frac{\partial \psi^*}{\partial y^*} = \frac{\partial \psi^*}{\partial \eta}\frac{\partial \eta}{\partial y^*} \tag{2.96a}$$
$$u^* = U^*_{ref} U_e F_\eta$$

$$v^* = -\frac{\partial \psi^*}{\partial x^*} = -\left(\frac{\partial \psi^*}{\partial \xi}\frac{\partial \xi}{\partial x^*} + \frac{\partial \psi^*}{\partial \eta}\frac{\partial \eta}{\partial x^*}\right) \tag{2.96b}$$
$$v^* = -U^*_{ref}\frac{U_e}{\sqrt{2\xi}}\left[F + 2\xi F_\xi + \left(\frac{2\xi \partial U_e/\partial x}{ReU_e^2} - 1\right)\eta F_\eta\right]$$

$$\frac{\partial u^*}{\partial x^*} = \frac{\partial u^*}{\partial x^*} + \frac{\partial u^*}{\partial \xi}\frac{\partial \xi}{\partial x^*} + \frac{\partial u^*}{\partial \eta}\frac{\partial \eta}{\partial x^*} \tag{2.96c}$$
$$\frac{\partial u^*}{\partial x^*} = \frac{U^*_{ref}}{L^*_{ref}}\left[\frac{\partial U_e}{\partial x}F_\eta + ReU_e^2 F_{\xi\eta} + \left(\frac{\partial U_e}{\partial x} - \frac{ReU_e^2}{2\xi}\right)\eta F_{\eta\eta}\right]$$

$$\frac{\partial u^*}{\partial y^*} = \frac{\partial u^*}{\partial \eta}\frac{\partial \eta}{\partial y^*} \tag{2.96d}$$
$$\frac{\partial u^*}{\partial y^*} = \frac{U^*_{ref}}{L^*_{ref}}\frac{ReU_e^2}{\sqrt{2\xi}}F_{\eta\eta}$$

$$\frac{\partial^2 u^*}{\partial y^{*2}} = \frac{\partial^2 u^*}{\partial \eta^2}\left(\frac{\partial \eta}{\partial y^*}^2\right) \tag{2.96e}$$
$$\frac{\partial^2 u^*}{\partial y^{*2}} = \frac{U^*_{ref}}{{L^*_{ref}}^2}\frac{Re^2U_e^3}{2\xi}F_{\eta\eta\eta}.$$

After inserting the derivatives (2.96) into the boundary-layer equation (2.25) the resulting partial differential equation with respect to ξ and η reads

$$F_{\eta\eta\eta} + FF_{\eta\eta} + \beta(\xi)\left(1 - F_\eta^2\right) = 2\xi\left(F_\eta F_{\xi\eta} - F_\xi F_{\eta\eta}\right) \tag{2.97}$$

where $\beta(\xi)$ is the dimensionless principal function [13] which is defined as

$$\beta(\xi) = \frac{2\partial U_e/\partial x}{U_e^2}\int_0^x U_e\,dx = \frac{2\xi\partial U_e/\partial x}{ReU_e^2}. \tag{2.98}$$

In the following work, equation (2.97) will be called *Görtler equation* and the corresponding transformations in equation (2.92) and (2.93) are denoted as *Görtler transformation*. The corresponding boundary conditions are found by inserting the transformations into the original boundary conditions (2.27) and those are found to be

$$\begin{aligned} \eta = 0: &\qquad F = 0 \quad \text{and} \quad F_\eta = 0 \\ \eta \to \infty: &\qquad F_\eta = 1. \end{aligned} \tag{2.99}$$

In the same procedure as for the Falkner-Skan transformation, a wedge flow (2.82) results in an ordinary differential equation of third order as the right-hand side of the Görtler equation cancels out as all derivatives with respect to ξ vanish. Therefore, the Görtler equation for a similarity solution is

$$F_{\eta\eta\eta} + FF_{\eta\eta} + \beta(\xi)\left(1 - F_\eta^2\right) = 0 \tag{2.100}$$

and the principal function β becomes the Hartree parameter (2.86).

By solving the boundary-layer equations with Görtler's transformation, the integral values δ_1 for the displacement thickness and δ_2 for the momentum thickness can be obtained in the following way

$$\delta_1^* = \frac{\nu^*\sqrt{2\xi}}{U_e^*}\eta_0(x) = L_{ref}^*\frac{\sqrt{2\xi}}{ReU_e}\eta_0(x) \qquad \text{with} \quad \eta_0(x) = \int_0^\infty F_\eta\left(1-F_\eta\right)d\eta = \lim_{\eta\to\infty}\left\{\eta - F(\xi,\eta)\right\} \tag{2.101}$$

$$\delta_2^* = \frac{\nu^*\sqrt{2\xi}}{U_e^*}\int_0^\infty F_\eta\left(1-F_\eta\right)d\eta = L_{ref}^*\frac{\sqrt{2\xi}}{ReU_e}\int_0^\infty F_\eta\left(1-F_\eta\right)d\eta. \tag{2.102}$$

Further, the friction coefficient C_f can be found which is characterized by the local shear stress at the wall. By using the definition (2.31) and taking the partial derivative $\frac{\partial u^*}{\partial y^*}$ from equation (2.96d) and evaluating at the wall ($\eta = 0$), the friction coefficient is given by

$$C_f = \frac{2U_e^2}{\sqrt{2\xi}}F_{\eta\eta}(\xi,0) \tag{2.103}$$

where U_∞^* is the free stream velocity at infinity.

2.6 Turbulence Models

In order to calculate the boundary layer after the transition point correctly, turbulence models have to be included in the boundary-layer equation (2.25). Therefore, the viscosity can be split into a viscosity term, ν^*, that is constant in laminar flows and a turbulent viscosity term, ν_t^* which is used to model the Reynolds stress term in the momentum equation. The turbulent viscosity is thus added to the laminar viscosity and modeled by the eddy viscosity approach. In order to keep the calculation simple a zero-equation model is implemented. The boundary-layer equation including the Reynolds shear stress term reads

$$u^*\frac{\partial u^*}{\partial x^*} + v^*\frac{\partial u^*}{\partial y^*} = U_e^*\frac{\partial U_e^*}{\partial x^*} + \frac{1}{\rho^*}\frac{\partial}{\partial y^*}\left(\mu^*\frac{\partial u^*}{\partial y^*} - \rho^*\overline{u^{*\prime}v^{*\prime}}\right) \tag{2.104}$$

where $-\rho^*\overline{u^{*\prime}v^{*\prime}}$ is known as the Reynolds shear stress.

2.6.1 Cebeci and Smith Turbulence Model

The Cebeci and Smith turbulence model [8] models the Reynolds shear stress by using an eddy viscosity approach. Thereby, the Reynolds shear stress is expressed as

$$-\rho^*\overline{u^{*\prime}v^{*\prime}} = \rho^*\nu_t^*\frac{\partial u^*}{\partial y^*} \tag{2.105}$$

where ν_t^* is the turbulent viscosity which has to be modeled. In the following the turbulent viscosity will be modeled for the inner and outer region of the boundary layer as two different behaviors are noticed. The inner region is

$$\nu_{t,i}^* = l^{*2}\left|\frac{\partial u^*}{\partial y^*}\right|\gamma_{tr} \quad \text{for} \quad 0 \le y^* \le y_c^* \tag{2.106}$$

where l^* is the mixing length, γ_{tr} is an intermittency factor and y_c^* the wall-normal distance where the outer region for the turbulent viscosity starts. The mixing length is expressed as

$$l^* = \kappa y^* \left[1 - e^{-\frac{y^*}{A^*}}\right] \tag{2.107}$$

where $\kappa = 0.4$ and A^* is a damping length constant which is given as

$$A^* = 26\frac{\nu^*}{N u_\tau^*} \quad \text{with} \quad N = \left(1 - 11.8p^+\right)^{1/2} \quad \text{and} \quad p^+ = \frac{\nu^* U_e^*}{{u_\tau^*}^3}\frac{dU_e^*}{dx^*}. \tag{2.108}$$

The friction velocity u_τ^* is defined as

$$u_\tau^* = \sqrt{\frac{\tau_w^*}{\rho^*}}. \tag{2.109}$$

Finally, the intermittency factor γ_{tr} ensures a smooth transition from a laminar to a fully turbulent flow. From the stagnation point to the onset of transition $\gamma_{tr} = 0$ and it smoothly increases during the transition process to $\gamma_{tr} = 1$ which then signifies that the flow is fully turbulent. In [8] the expression is given as

$$\gamma_{tr} = 1 - e^{-G\left(x^* - x_{tr}^*\right)\int_{x_{tr}^*}^{x^*}\frac{dx^*}{U_e^*}} \tag{2.110}$$

with

$$G = \left(\frac{3}{C^2}\right)\left(\frac{U_e^{*3}}{\nu^{*2}}\right) Re_{x_{tr}}^{-1.34} \quad \text{with} \quad C^2 = 213\left(\log Re_{x_t r} - 4.7323\right). \tag{2.111}$$

The range of validity for C^2 is for $2.4 \times 10^5 \leq Re_L \leq 2 \times 10^6$.
The turbulent viscosity in the outer region reads [8]

$$\nu_{t,o}^* = \alpha U_e^* \delta_1^* \gamma_{tr} \gamma \quad \text{for} \quad y_c^* \leq y^* \leq \delta^* \tag{2.112}$$

where $\alpha = 0.0168$ and γ is an intermittency factor for the outer region expressed as

$$\gamma = \left[1 + 5.5\left(\frac{y^*}{\delta^*}\right)^6\right]^{-1}. \tag{2.113}$$

Equations (2.106) to (2.113) are used to compute the turbulent viscosity ν_t^* for the whole region.
The turbulent boundary-layer equation in dimensionless form then becomes

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = U_e\frac{\partial U_e}{\partial x} + \frac{1}{Re}\frac{\partial}{\partial y}\left[\left(1 + \frac{\nu_t^*}{\nu^*}\right)\frac{\partial u}{\partial y}\right]. \tag{2.114}$$

Another formula for the mixing length for the whole layer is introduced by Michel [8]. It is valid for flows with no mass transfer and smooth surfaces. These assumptions apply for the work here. The mixing length is found to be

$$l^* = \delta^*\left[1 - e^{\frac{y^*}{A^*}}\right] 0.085 \tanh\left(\frac{\kappa}{0.085}\frac{y^*}{\delta^*}\right) \tag{2.115}$$

with $\kappa = 0.41$ and $A^* = 26\nu^*/u_\tau^*$. Thereby, the turbulent viscosity reads

$$\nu_t^* = {l^*}^2\frac{\partial u^*}{\partial y^*}. \tag{2.116}$$

2.7 Inverse Boundary-Layer Method

As already mentioned above the direct boundary-layer equation (2.25) cannot be solved for separating flows due to the Goldstein singularity [12]. Goldstein showed that at the separation point the pressure distribution cannot be taken arbitrarily but it has fulfill several conditions. In the following, the solution of the boundary-layer equation with the help of a given pressure or edge velocity distribution will be called the *standard problem*. In order to find a solution that does not break down at the separation point an *inverse method* has to be found which does not depend on the edge velocity.

2.7.1 Method of Catherall and Mangler

In order to overcome the breakdown of the numerical scheme at the separation point Catherall and Mangler [4] introduced a scheme that iterates the edge velocity as part of the solution. Thereby, they ensured that the singularity does not occur and the marching scheme does not diverge at the separation point. In order to determine the edge velocity the displacement thickness has to be known which is obtained by the *standard method* for all locations upstream the separation point. At the separation point and furhter downstream the displacement thickness has to be extrapolated. By knowing the displacement thickness distribution the edge velocity of the viscous method can be obtained by applying an inverse scheme. In order to compute an updated inviscid velocity distribution, the displacement thickness is added on top of the airfoil to include the viscous effects. Then, the inviscid solver is used to calculate a new edge velocity distribution. Both velocities are then used to compute an updated displacement thickness distribution

$$\delta_1 = \delta_1^0 \left[1 + \omega \left(\frac{U_{e,v}}{U_{e,i}} - 1 \right) \right] \tag{2.117}$$

where δ_1^0 is a given distribution for the displacement thickness (extrapolated after the separation point), ω is a relaxation parameter and $U_{e,v}$ and $U_{e,i}$ are the edge velocity distributions for the viscous and inviscid solver, respectively. After having obtained a new distribution for the displacement thickness with equation (2.117) the calculation starts again. The solution is converged if the change of the displacement thickness is less than a given tolerance level.
The choice of the relaxation factor distinguishes if the scheme converges or diverges but there is always some arbitrariness in selecting the relaxation factor. Depending on the implementation under-relaxation might be needed to derive a converging scheme and thus the solution might converge only slowly.

2.7.2 Method of Veldman

In 1981 Veldman [25] came up with a new approach solving the boundary layer for the displacement thickness and the edge velocity simultaneously. Thereby, a stable scheme is found that works with an over-relaxation factor of 1.5 and thus allows quick convergence.
Veldman used triple-deck theory to derive his approach. Triple-deck theory is useful to determine the character of the flow and thus to find the governing equations for each layer. The triple-deck separates the flow into three different layers normal to the wall. In incompressible flow the extent in streamwise direction is $\mathcal{O}(Re^{-3/8}L^*)$ around the singular point [25]. In Figure 2.7 the triple-deck is sketched.
The lower first two layers are sufficiently described by the boundary-layer equation derived in section 2.2. The lowest layer represents the viscous part of the flow having a thickness of $\mathcal{O}(Re^{-5/8}L^*)$ whereas

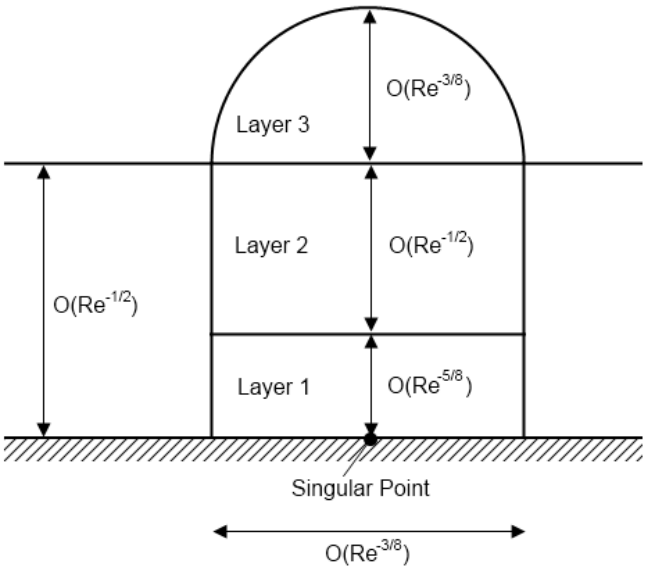


Figure 2.7.: Triple-Deck.

the middle layer (thickness $\mathcal{O}(Re^{-1/2}L^*)$) is considered inviscid. The outer layer of thickness $\mathcal{O}(Re^{-3/8}L^*)$ represents the region where the character of the flow changes rapidly. Nevertheless, the interaction has only local character due to the small extent of $\mathcal{O}(Re^{-3/8}L^*)$. Far away from the separation point the pressure is determined by the inviscid region and thus the standard method can be used to solve the flow. However, if the flow approaches the separation point the character changes and the pressure can be determined by solving the boundary-layer equation. This requires the solution with an inverse solver. In order to make use of the findings the edge velocity distribution is described as in [25]

$$U_e(x) = U_e^0(x) + \delta U_e(x) \quad \text{with} \quad \delta U_e(x) = \frac{1}{\pi} \int_{x_a}^{x_b} \frac{d}{d\sigma} \left(U_e \delta_1\right) \frac{d\sigma}{x-\sigma} \tag{2.118}$$

where $U_e^0(x)$ is the edge velocity distribution from the inviscid solver and $\delta U_e(x)$ the perturbation velocity. The latter accounts for the change of δ_1, also in the region $[x_a, x_b]$ where the singularity and thus the change in character of the flow occurs.
The perturbation velocity $\delta U_e(x)$ is described by a similar ansatz as used in thin airfoil theory accounting for the deflection of the streamlines of the outer flow [2]. As illustrated in Figure 2.7 the boundary layer effects the outer flow in the region of the separation point. With the ansatz given in (2.118), the displacement thickness and the edge velocity are used to solve the boundary-layer equation. Thereby, the Veldman method accounts for the change of character of the flow at the separation point.
In inverse form the Falkner-Skan transformation described in section 2.5.1 is calculated with U_∞^* (the reference velocity) because the edge velocity is unknown a priori. The transformation variable in wall-normal direction and the resulting stream function then read

$$Y = y^* \sqrt{\frac{U_\infty^*}{\nu^* x^*}} = y \sqrt{\frac{Re}{x}} \tag{2.119}$$

$$\psi^* = \sqrt{U_\infty^* \nu^* x^*} F(x^*, Y) = U_{ref}^* L_{ref}^* \sqrt{\frac{x}{Re}} F(x, Y). \tag{2.120}$$

The function F is a dimensionless stream function that depends on the streamwise coordinate and the wall-normal coordinate Y that is defined with the free-stream velocity at infinity. Note that the wall-normal coordinate formed with the local edge velocity is denoted by η as given in equation (2.71). That

variable is used to solve the standard problem. If the wall-normal coordinate Y is used, the problem has to be solved in inverse mode.
Using those functions given in equation (2.119) and (2.120) analogous to the derivation in the Falkner-Skan section 2.5.1, the boundary-layer equation (2.25) can be written as

$$F_{YYY} + \frac{1}{2} F F_{YY} = x \left(F_Y \frac{\partial F_Y}{\partial x} - F_{YY} \frac{\partial F}{\partial x} \right) - x U_e \frac{dU_e}{dx}. \tag{2.121}$$

2.8 Transition Prediction by Means of Stability Analysis

The basic idea in linear stability analysis is to disturb a known mean flow with small perturbations. The mean flow is obtained by solving the boundary-layer equation. In the following, those disturbances will be represented by normal modes and the flow can be analyzed by solving an eigenvalue problem which shows if the disturbances are amplified or damped. In order to derive the equations which lead to the eigenvalue problem, the three-dimensional, incompressible continuity equation and Navier-Stokes equations (2.5) and (2.6) are converted into dimensionless equations with the transformations given in equation (2.13). In vector form those equations read

$$\nabla \cdot \vec{u} = 0 \tag{2.122}$$

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} = -\nabla p + \frac{1}{Re} \nabla^2 \vec{u} \tag{2.123}$$

where $\vec{u} = \vec{U} + \vec{u}'$ is a composition of the mean flow $\vec{U}$ found by solving the boundary-layer equations and the perturbation velocity $\vec{u}'$. Similarly, $p = P + p'$ is a composition of the mean pressure P and the perturbation pressure p'. In the most general approach the disturbances can have any form. Due to the fact that the mean flow fulfills the continuity equation and the Navier-Stokes equation those values can be subtracted from (2.122) and (2.123). Further, the perturbations are assumed to be small and thus products of perturbation terms are neglected. Thereby, the linearized set of equations in vector form follows

$$\nabla \cdot \vec{u}' = 0 \tag{2.124}$$

$$\frac{\partial \vec{u}'}{\partial t} + \vec{u}' \cdot \nabla \vec{U} + \vec{U} \cdot \nabla \vec{u}' = -\nabla p' + \frac{1}{Re} \nabla^2 \vec{u}'. \tag{2.125}$$

In component notation which is here introduced due to reasons of clarity equations (2.124) and (2.125) read

$$\frac{\partial u'}{\partial x} + \frac{\partial v'}{\partial y} + \frac{\partial w'}{\partial z} = 0 \tag{2.126}$$

and

$$\frac{\partial u'}{\partial t}+u'\frac{\partial U}{\partial x}+v'\frac{\partial U}{\partial y}+w'\frac{\partial U}{\partial z}+ \\ +U\frac{\partial u'}{\partial x}+V\frac{\partial u'}{\partial y}+W\frac{\partial u'}{\partial z}=-\frac{\partial p'}{\partial x}+\frac{1}{Re}\left(\frac{\partial^2 u'}{\partial x^2}+\frac{\partial^2 u'}{\partial y^2}+\frac{\partial^2 u'}{\partial z^2}\right) \tag{2.127a}$$

$$\frac{\partial v'}{\partial t}+u'\frac{\partial V}{\partial x}+v'\frac{\partial V}{\partial y}+w'\frac{\partial V}{\partial z}+ \\ +U\frac{\partial v'}{\partial x}+V\frac{\partial v'}{\partial y}+W\frac{\partial v'}{\partial z}=-\frac{\partial p'}{\partial y}+\frac{1}{Re}\left(\frac{\partial^2 v'}{\partial x^2}+\frac{\partial^2 v'}{\partial y^2}+\frac{\partial^2 v'}{\partial z^2}\right) \tag{2.127b}$$

$$\frac{\partial w'}{\partial t}+u'\frac{\partial W}{\partial x}+v'\frac{\partial W}{\partial y}+w'\frac{\partial W}{\partial z}+ \\ +U\frac{\partial w'}{\partial x}+V\frac{\partial w'}{\partial y}+W\frac{\partial w'}{\partial z}=-\frac{\partial p'}{\partial z}+\frac{1}{Re}\left(\frac{\partial^2 w'}{\partial x^2}+\frac{\partial^2 w'}{\partial y^2}+\frac{\partial^2 w'}{\partial z^2}\right). \tag{2.127c}$$

Those equations are the basis for the derivation of the stability equations.

2.8.1 Transition Prediction by Local Stability Analysis

In this section the foundation for local stability analysis is derived, namely, the local Orr-Sommerfeld and Squire equations. Therefore, a parallel mean flow in x-direction will be considered with derivatives that only depend on the wall-normal direction, e.g. $\vec{\mathbf{U}} = (U(y), 0, 0)^T$. In boundary-layer flow this is only an approximation since there is a usually small flow component into the wall-normal direction, which is responsible for the boundary-layer growth. Introducing such a flow to the general set of equations (2.126) and (2.127) the reduced set of equations

$$\frac{\partial u'}{\partial x}+\frac{\partial v'}{\partial y}+\frac{\partial w'}{\partial z}=0 \tag{2.128}$$

and

$$\frac{\partial u'}{\partial t}+v'\frac{\partial U}{\partial y}+U\frac{\partial u'}{\partial x}=-\frac{\partial p'}{\partial x}+\frac{1}{Re}\left(\frac{\partial^2 u'}{\partial x^2}+\frac{\partial^2 u'}{\partial y^2}+\frac{\partial^2 u'}{\partial z^2}\right) \tag{2.129a}$$

$$\frac{\partial v'}{\partial t}+U\frac{\partial v'}{\partial x}=-\frac{\partial p'}{\partial y}+\frac{1}{Re}\left(\frac{\partial^2 v'}{\partial x^2}+\frac{\partial^2 v'}{\partial y^2}+\frac{\partial^2 v'}{\partial z^2}\right) \tag{2.129b}$$

$$\frac{\partial w'}{\partial t}+U\frac{\partial w'}{\partial x}=-\frac{\partial p'}{\partial z}+\frac{1}{Re}\left(\frac{\partial^2 w'}{\partial x^2}+\frac{\partial^2 w'}{\partial y^2}+\frac{\partial^2 w'}{\partial z^2}\right) \tag{2.129c}$$

is found. In order to find an expression for the pressure, the divergence of the Navier-Stokes equation (2.129) results in

$$\nabla^2 p'+2U_y v'_x+\frac{\partial}{\partial t}\left(u'_x+v'_y+w'_z\right)+U\frac{\partial}{\partial x}\left(u'_x+v'_y+w'_z\right)- \\ -\frac{1}{Re}\left[\frac{\partial^2}{\partial x^2}\left(u'_x+v'_y+w'_z\right)+\frac{\partial^2}{\partial y^2}\left(u'_x+v'_y+w'_z\right)+\frac{\partial^2}{\partial z^2}\left(u'_x+v'_y+w'_z\right)\right]=0$$

which reduces in combination with the continuity equation (2.128) to

$$\nabla^2 p' = -2U_y v'_x. \tag{2.130}$$

The subscripts mean derivatives with respect to the variable given as subscript like already denoted in the previous sections. The Orr-Sommerfeld equation is found by taking the Laplacian of equation (2.129b)

$$\frac{\partial}{\partial t}\nabla^2 v' + U\frac{\partial}{\partial x}\nabla^2 v' + U_{yy}v'_x + 2U_y v'_{xy} + \frac{\partial}{\partial y}\nabla^2 p' - \frac{1}{Re}\nabla^4 v' = 0$$

and using equation (2.130) to replace the pressure which results in

$$\left[\left(\frac{\partial}{\partial t} + U\frac{\partial}{\partial x}\right)\nabla^2 - \frac{\partial^2 U}{\partial y^2}\frac{\partial}{\partial x} - \frac{1}{Re}\nabla^4\right] v' = 0. \tag{2.131}$$

This equation is employed to find the wall-normal velocity perturbation v'. In order to describe the disturbances in all three dimensions another equation has to be derived. Schmid and Henningson [21] utilized the wall-normal vorticity to find a convenient equation. Again, only the disturbances are interesting as the base flow must fulfill the vorticity equation which then simplifies for the wall-normal direction to

$$\Omega'_y = \frac{\partial u'}{\partial z} - \frac{\partial w'}{\partial x}. \tag{2.132}$$

By differentiating the Navier-Stokes equation in x-direction (2.129a) with respect to z and the z-component of the Navier-Stokes equation (2.129c) with respect to x and subtracting them from each other the pressure term cancels out and it follows

$$u'_{zt} - w'_{xt} + U_y v'_z + U u'_{xz} - U w'_{xx} - \frac{1}{Re}\left[\frac{\partial}{\partial z}\left(u'_{xx} + u'_{yy} + u'_{zz}\right) - \frac{\partial}{\partial x}\left(u'_{xx} + u'_{yy} + u'_{zz}\right)\right] = 0$$

which can also be written as

$$\begin{aligned}\frac{\partial}{\partial t}\left(u'_z - w'_x\right) + U\frac{\partial}{\partial x}\left(u'_z - w'_x\right) - \\ -\frac{1}{Re}\left[\frac{\partial^2}{\partial x^2}\left(u'_z - w'_x\right) + \frac{\partial^2}{\partial y^2}\left(u'_z - w'_x\right) + \frac{\partial^2}{\partial z^2}\left(u'_z - w'_x\right)\right] = -U_y v'_z.\end{aligned}$$

By using the identity (2.132) a second equation is found

$$\left[\frac{\partial}{\partial t} + U\frac{\partial}{\partial x} - \frac{1}{Re}\nabla^2\right]\Omega'_y = -\frac{\partial U}{\partial y}\frac{\partial v'}{\partial z}. \tag{2.133}$$

It describes the disturbances in all three dimensions in combination with equation (2.131). Note that the right-hand side of equation (2.133) is the coupling term which connects both equations. The following boundary conditions apply to both equations:

$$\begin{aligned} y = 0: &\qquad v' = 0, \quad \frac{\partial v'}{\partial y} = 0 \quad \text{and} \quad \Omega'_y = 0 \\ y \to \infty: &\qquad v' = 0, \quad \frac{\partial v'}{\partial y} = 0 \quad \text{and} \quad \Omega'_y = 0. \end{aligned} \tag{2.134}$$

Equations (2.131), (2.133) and (2.134) are valid for a parallel base flow that only depends on the wall-normal coordinate and an arbitrary disturbance in space and time.
In the following, the disturbances are expressed as normal modes. A wavelike solution is assumed having the form

$$v'(x,y,z,t) = \hat{v}(y)\exp^{i(\alpha x+\beta z-\omega t)} \tag{2.135a}$$

$$\Omega'_y(x,y,z,t) = \hat{\Omega}_y(y)\exp^{i(\alpha x+\beta z-\omega t)} \tag{2.135b}$$

where α and β are the spanwise and streamwise wave numbers, respectively, and ω is the angular frequency. The shape functions are given by $\hat{v}$ and $\hat{\Omega}_y$. Using that disturbance ansatz and inserting it into equations (2.131) and (2.133) results in the following system

$$\left[(-i\omega + i\alpha U)\left(\frac{\partial^2}{\partial y^2} - k^2\right) - i\alpha\frac{\partial^2 U}{\partial y^2} - \frac{1}{Re}\left(k^2 - \frac{\partial^2}{\partial y^2}\right)^2\right]\hat{v} = 0 \tag{2.136}$$

$$\left[-i\omega + i\alpha U - \frac{1}{Re}\left(\frac{\partial^2}{\partial y^2} - k^2\right)\right]\hat{\Omega}_y = -i\beta\frac{\partial U}{\partial y}\hat{v} \tag{2.137}$$

where $k = \sqrt{\alpha^2+\beta^2}$ is the wave number. The boundary conditions (2.134) apply to that case as well. The first equation (2.136) is commonly known as the *Orr-Sommerfeld equation* and the second one is the *Squire equation* which is needed to capture three-dimensional disturbances. If the spanwise wave number β is zero, the Squire equation is not needed. Another fact that is worth to mention here is the reduction from four equations in (2.128) and (2.129) to only two equations.
In the following a different derivation for the Orr-Sommerfeld (2.136) and the Squire equation (2.137) will be mentioned which is used for numerical solution later on. Therefore, the disturbance ansatz

$$\boldsymbol{q}'(x,y,z,t) = \hat{\boldsymbol{q}}(y)\exp^{i\left(\alpha x+\beta z-\omega t\right)} \tag{2.138}$$

is chosen where $\hat{\boldsymbol{q}} = (\hat{u},\hat{v},\hat{w},\hat{p})^T$ is the shape function, α is the chordwise wavenumber, β is the spanwise wavenumber and ω is the angular frequency. Inserting that ansatz into equations (2.128) and (2.129) yields

$$i\alpha\hat{u} + \hat{v}_y + i\beta\hat{w} = 0 \tag{2.139a}$$

$$-i\omega\hat{u} + \hat{v}U_y + iU\alpha\hat{u} + i\alpha\hat{p} + \frac{1}{Re}\left(\alpha^2\hat{u} + \beta^2\hat{u} - \hat{u}_{yy}\right) = 0 \tag{2.139b}$$

$$-i\omega\hat{v} + iU\alpha\hat{v} + \hat{p}_y + \frac{1}{Re}\left(\alpha^2\hat{v} + \beta^2\hat{v} - \hat{v}_{yy}\right) = 0 \tag{2.139c}$$

$$-i\omega\hat{w} + iU\alpha\hat{w} + i\beta\hat{p} + \frac{1}{Re}\left(\alpha^2\hat{w} + \beta^2\hat{w} - \hat{w}_{yy}\right) = 0. \tag{2.139d}$$

Note that still four equations have to be solved and that the number is not reduced to two as above. However, the order of differentiation is only second order whereas the equations derived before are up to fourth order. Those equations can be converted into matrix form and a linear operator can be set. The operator will be similar to the one derived for the parabolized stability equations in the next section. The system of equations in matrix form in which the local stability equations (2.139) could be transferred as well is given in equation (2.144) even though some terms do not occur as they are excluded due to the assumptions of local stability theory.

2.8.2 Transition Prediction by Parabolized Stability Equations

In contrast to the local stability analysis presented above the non-parallel effects can be taken into account if parabolized stability equations are used. The parabolized stability equations were originally derived by Herbert and Bertolotti in 1987 [15]. In the following, the parabolized stability equations are declared for the general case of a three-dimensional mean flow that is constant in spanwise direction. Therefore, the derivatives $\frac{\partial U}{\partial z} = \frac{\partial V}{\partial z} = \frac{\partial W}{\partial z} = 0$ and the linearized continuity equation and Navier-Stokes equations (2.126) and (2.127) containing disturbances and mean flow values reduce to

$$\frac{\partial u'}{\partial x} + \frac{\partial v'}{\partial y} + \frac{\partial w'}{\partial z} = 0 \tag{2.140}$$

and

$$\frac{\partial u'}{\partial t} + u'\frac{\partial U}{\partial x} + v'\frac{\partial U}{\partial y} + U\frac{\partial u'}{\partial x} + V\frac{\partial u'}{\partial y} + W\frac{\partial u'}{\partial z} = -\frac{\partial p'}{\partial x} + \frac{1}{Re}\left(\frac{\partial^2 u'}{\partial x^2} + \frac{\partial^2 u'}{\partial y^2} + \frac{\partial^2 u'}{\partial z^2}\right) \tag{2.141a}$$

$$\frac{\partial v'}{\partial t} + u'\frac{\partial V}{\partial x} + v'\frac{\partial V}{\partial y} + U\frac{\partial v'}{\partial x} + V\frac{\partial v'}{\partial y} + W\frac{\partial v'}{\partial z} = -\frac{\partial p'}{\partial y} + \frac{1}{Re}\left(\frac{\partial^2 v'}{\partial x^2} + \frac{\partial^2 v'}{\partial y^2} + \frac{\partial^2 v'}{\partial z^2}\right) \tag{2.141b}$$

$$\frac{\partial w'}{\partial t} + u'\frac{\partial W}{\partial x} + v'\frac{\partial W}{\partial y} + U\frac{\partial w'}{\partial x} + V\frac{\partial w'}{\partial y} + W\frac{\partial w'}{\partial z} = -\frac{\partial p'}{\partial z} + \frac{1}{Re}\left(\frac{\partial^2 w'}{\partial x^2} + \frac{\partial^2 w'}{\partial y^2} + \frac{\partial^2 w'}{\partial z^2}\right). \tag{2.141c}$$

The disturbances are assumed to have the following form

$$\boldsymbol{q}'(x,y,z,t) = \hat{\boldsymbol{q}}(x,y)\exp^{i\left(\int_{x_0}^{x}\alpha(\xi)d\xi + \beta z - \omega t\right)} \tag{2.142}$$

where $\hat{\boldsymbol{q}} = (\hat{u},\hat{v},\hat{w},\hat{p})^T$, α is the chordwise wavenumber, β is the spanwise wavenumber and ω is the angular frequency. Note that $\hat{\boldsymbol{q}}(x,y)$ is called shape function which is supposed to capture the slowly varying shape of the disturbances $\boldsymbol{q}'$ whereas the exponential part of (2.142) represents the quickly oscillating wavelike behavior. In contrast to the local stability equations described in section 2.8.1 the disturbances also depend on the streamwise location x. Further, the chordwise wavenumber is integrated along the region of interest and thus the dependence of the streamwise change of the flow is included. In the next step the disturbances (2.142) are inserted into equations (2.140) and (2.141) which read

$$\hat{u}_x + i\alpha\hat{u} + \hat{v}_y + i\beta\hat{w} = 0 \tag{2.143a}$$

$$-i\omega\hat{u} + \hat{u}U_x + \hat{v}U_y + U\hat{u}_x + iU\alpha\hat{u} + V\hat{u}_y + iW\beta\hat{u} + (\hat{p}_x) + i\alpha\hat{p} + \frac{1}{Re}\left(\alpha^2\hat{u} + \beta^2\hat{u} - \hat{u}_{yy}\right) = 0 \tag{2.143b}$$

$$-i\omega\hat{v} + \hat{u}V_x + \hat{v}V_y + U\hat{v}_x + iU\alpha\hat{v} + V\hat{v}_y + iW\beta\hat{v} + \hat{p}_y + \frac{1}{Re}\left(\alpha^2\hat{v} + \beta^2\hat{v} - \hat{v}_{yy}\right) = 0 \tag{2.143c}$$

$$-i\omega\hat{w} + (\hat{u}W_x) + \hat{v}W_y + U\hat{w}_x + iU\alpha\hat{w} + V\hat{w}_y + iW\beta\hat{w} + i\beta\hat{p} + \frac{1}{Re}\left(\alpha^2\hat{w} + \beta^2\hat{w} - \hat{w}_{yy}\right) = 0 \tag{2.143d}$$

where subscripts denote derivations with respect to the corresponding direction. Note that in equations (2.143) higher order terms are omitted. The scaling that are applied to the find out the order of the terms can be found in [23]. Further, in equation (2.143b) the term $\hat{p}_x$ exhibits a slight elliptic behavior. Therefore, as suggested by Tempelmann et al. [23] the term $\hat{p}_x$ is omitted in order to relax the equation

to improve numerical stability. Furthermore, the term $\hat{u}W_x$ in equation (2.143d) is omitted as it was only found to be important for non-modal growth as suggested by Tempelmann et al. [23]. Due to reasons of clearness the stability equations (2.143) are expressed in operator form as

$$\boldsymbol{L}\hat{\boldsymbol{q}} = 0 \tag{2.144}$$

with the linear operator

$$\boldsymbol{L} = \boldsymbol{A} + \boldsymbol{B}\frac{\partial}{\partial y} + \boldsymbol{C}\frac{\partial^2}{\partial y^2} + \boldsymbol{D}\frac{\partial}{\partial x}. \tag{2.145}$$

The matrices $\boldsymbol{A}$ till $\boldsymbol{D}$ are then found to be

$$\boldsymbol{A} = \begin{pmatrix} i\alpha & 0 & i\beta & 0 \\ C + U_x & U_y & 0 & i\alpha \\ V_x & C + V_y & 0 & 0 \\ 0 & W_y & C & i\beta \end{pmatrix} \tag{2.146a}$$

$$\boldsymbol{B} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ V & 0 & 0 & 0 \\ 0 & V & 0 & 1 \\ 0 & 0 & V & 0 \end{pmatrix} \tag{2.146b}$$

$$\boldsymbol{C} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -\frac{1}{Re} & 0 & 0 & 0 \\ 0 & -\frac{1}{Re} & 0 & 0 \\ 0 & 0 & -\frac{1}{Re} & 0 \end{pmatrix} \tag{2.146c}$$

$$\boldsymbol{D} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ U & 0 & 0 & 0 \\ 0 & U & 0 & 0 \\ 0 & 0 & U & 0 \end{pmatrix} \tag{2.146d}$$

where

$$C = i\omega + iU\alpha + iW\beta + \frac{1}{Re}\left(\alpha^2 + \beta^2\right). \tag{2.147}$$

In order to ensure that the assumption that the amplitude functions $\hat{\boldsymbol{q}}$ only vary slowly in x-direction is valid an auxiliary function

$$\int_0^\infty \hat{\boldsymbol{q}}^H \frac{\partial}{\partial x}\hat{\boldsymbol{q}}\, dy = 0 \tag{2.148}$$

is introduced. The superscript ^{H} denotes the complex conjugate transpose. That function ensures that the slow variations are captured by the shape function whereas the fast oscillations are captured in the exponential part of the disturbance ansatz (2.142).

3 Implementation of Laminar Boundary-Layer Methods

In the following chapter the solution of the laminar boundary-layer equation is presented. Before the implementation is presented some general features are revealed at the beginning of the chapter in section 3.1. The implementation of the grid and the relevant nomenclature is given in section 3.1.1. Further, the calculation of the stagnation point which is needed as an input for all boundary calculations is presented in section 3.1.2. A convenient way to calculate derivatives by matrix multiplication is given in section 3.1.3. Besides of that a technique to solve a block tridiagonal matrix which is also build-in into *MATLAB* is shown in section 3.1.4.
In section 3.2 the integral boundary-layer equations from section 2.4 are solved by implementing the relevant equations into *MATLAB*. These solutions are compared with *XFOIL*, a powerful tool to design airfoils. *XFOIL* uses a Hess-Smith panal method to compute the pressure distribution and a integral method to compute the boundary-layer characteristics. Furthermore, *XFOIL* employs a viscous-inviscid interaction method to update the pressure distribution and thus considering the effects of the boundary layer. After solving the integral boundary-layer equations, in section 3.3 boundary-layer profiles are solved for similarity flows. These solutions are important as they are used as an input for the solution of boundary-layer for airfoil flows.
Finally, the solution of the boundary-layer profiles for airfoil flows is presented in section 3.4. The implementation of the boundary-layer equation is shown for two different discretization techniques. The transformations described in section 2.5 are implemented. The results of the calculations are then compared to a compressible boundary solver from KTH. The boundary-layer characteristics are also compared to *XFOIL*. The solution method presented in that section is referred to as *standard method*.
For the implementation of the integral boundary-layer equation and the boundary-layer equation finite differences are used to solve the equations. The initial pressure distribution to start the calculation is taken from an inviscid calculation from *XFOIL*.

3.1 Numerical Considerations

This section reveals some general numerical considerations that are important to solve the boundary-layer equations.

3.1.1 The Boundary-Layer Grid

In the following section the definition and creation of the grid is described. Similar to the definition in sections 2.5.1 and 2.5.2 an arbitrarily spaced grid in streamwise direction

$$\begin{aligned} &\xi^{(0)} = \xi_{start}; \\ &\xi^{(n)} = \xi^{(n-1)} + k^{(n)} \quad \text{for } n = 1, 2, \ldots, N \quad \text{and } \xi_{(N)} \equiv \xi_{end} \end{aligned} \tag{3.1}$$

and wall-normal direction η

$$\begin{aligned} &\eta_{(0)} = 0; \\ &\eta_{(j)} = \eta_{(j-1)} + h_{(j)} \quad \text{for } j = 1, 2, \ldots, J \quad \text{and } \eta_{(J)} \equiv \eta_\infty \end{aligned} \tag{3.2}$$

is introduced. The variables k and h denote the spacing in streamwise and wall-normal direction whereas the superscript n and the subscript j represent the current position in streamwise and wall-normal direction. The capital letter N denotes the total number of grid points in streamwise direction whereas J represents the total number of grid points in wall-normal direction and marks at the same time the numerical position of η_∞. Of course, in numerics η_∞ will be approximated with a maximum value η_{max} that stands for the edge of the boundary layer. The boundary-layer thickness δ may be characterized by 99% of the edge velocity at the current streamwise position. An illustration of a typical numerical grid is shown in Figure 3.1.

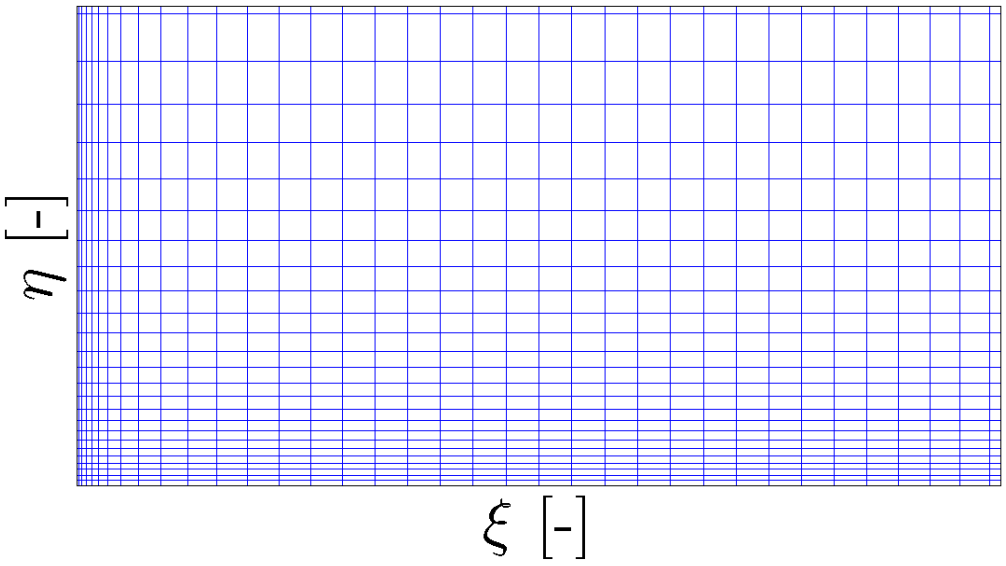


Figure 3.1.: The numerical grid with the characteristics of a arbitrarily spacing in ξ-direction and a non-uniform spacing defined by the grid parameter K in η-direction.

In the following no specific grid spacing in streamwise direction is used but the grid spacing from the potential flow solver. In wall-normal direction a very efficient grid spacing is introduced that becomes finer while approaching to the wall. The reason for that is to ensure the accuracy of the derivatives, which changes dramatically at the wall and become constant as the flow approaches the boundary-layer edge. Thereby, the number of grid points is reduced and numerical accuracy is obtained.
A suggestion of Cebeci [5] is used to generate a grid in wall-normal direction. The ratios of spacings between any two subsequent intervals, K, will be constant and the parameter will be referred to as *Variable Grid Parameter*. If the distance between the wall and the first grid point away from the wall, h_1, is given the grid can be generated in the following way

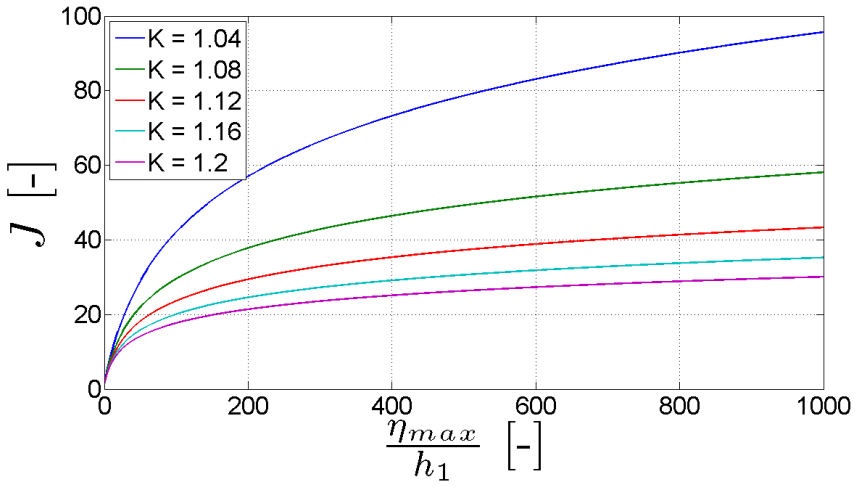
$$\eta_{(j)} = h_1 \frac{K^j - 1}{K - 1} \quad \text{for} \quad j = 1, 2, \ldots, J \quad \text{and} \quad K > 1 \tag{3.3a}$$

$$\eta_{(j)} = j h_1 \quad \text{for} \quad j = 1, 2, \ldots, J \quad \text{and} \quad K = 1 \tag{3.3b}$$

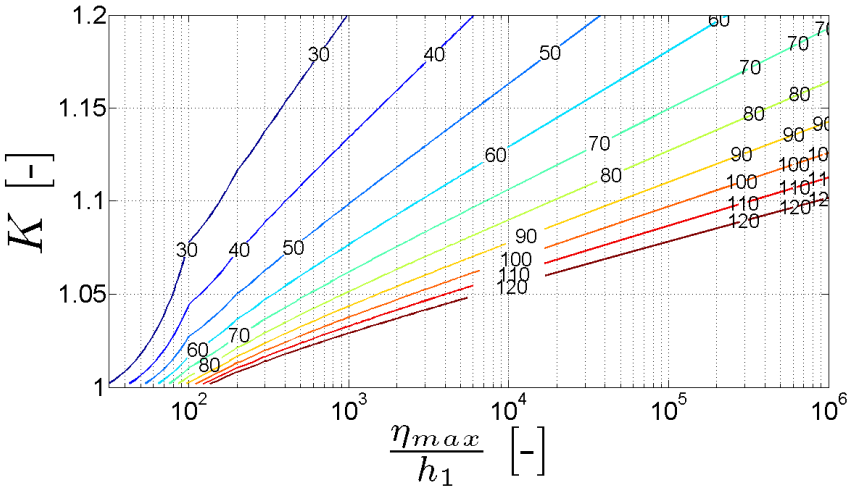
where the case $K = 1$ refers to an equidistant grid. The total number of grid points can be determined as

$$J = \frac{\ln\left[1 + (K-1)\frac{\eta_{max}}{h_1}\right]}{\ln K} + 1. \tag{3.4}$$

Cebeci [5] suggests values for h_1 and K to be 0.01 and 1.10, respectively. For laminar flows $\eta_{max} = 8$ should be sufficient whereas in turbulent flows and at higher Reynolds numbers the transformed boundary-layer thickness has to be increased. For the latter cases the first grid spacing should be chosen smaller and the variable grid parameter may be larger. The variation in the number of grid points needed for values of h_1, η_{max}, K and J is shown in Figure 3.2a and Figure 3.2b. Those figures can also be employed to chose values for J and K depending on the grid needed to resolve the flow.



(a) Variation of K with given values for h_1, η_{max} and J



(b) Total number of grid points J in dependence of h_1, η_{max} and K

Figure 3.2.: Effect of grid parameters.

The subroutine that is used to create the wall-normal grid points is implemented in `creategrid.m`.

3.1.2 Finding the Stagnation Point

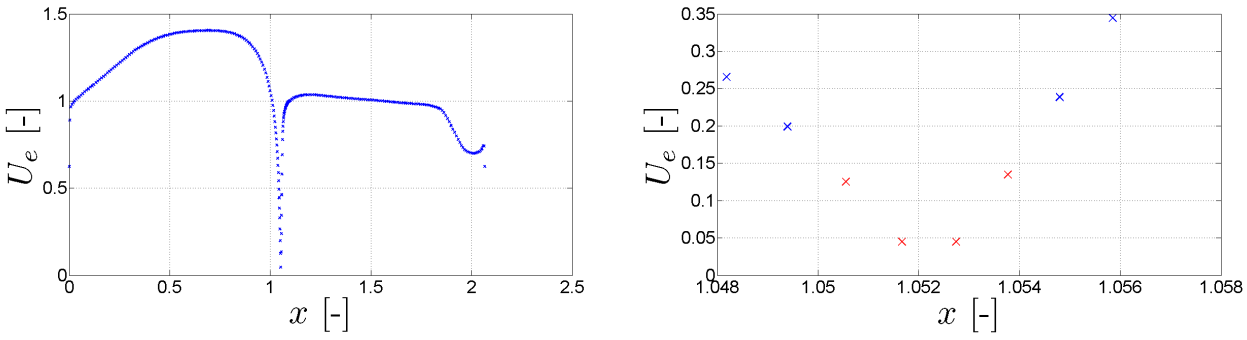
The method which is implemented in the following is based on linear extrapolation of the discrete solution points of the potential flow that are closest to the stagnation point. Specifically, the measured or computed pressure distribution is converted into a velocity distribution with

$$C_p = 1 - U_e^2 \quad \Leftrightarrow \quad U_e = \sqrt{1 - C_p} \tag{3.5}$$

which is derived from Bernoulli's equation. This conversion is valid for steady and incompressible flows. Further, the airfoil coordinates x_c and y_c are used to calculate the arc length of the airfoil with the formula of Pythagoras

$$x^{(n)} = x^{(n-1)} + \sqrt{\left(x_c^{(n-1)} - x_c^{(n-1)}\right)^2 + \left(y_c^{(n)} - y_c^{(n-1)}\right)^2} \quad \text{for} \quad n = 1, 2, \ldots, N \tag{3.6}$$

where x is the arc length and N is the total number of profile coordinates. Note that $x^{(0)} = 0$ is the starting value. The airfoil has to be discretized from the trailing edge over the suction side and the pressure side back to the trailing edge or the other way around. Those values can be correlated to the velocity distribution which then can be plotted against the arc length as depicted in Figure 3.3a. The two locations with the lowest edge velocity value need to be found and the two additional values before and after the two minimal values are used for the extrapolation. The selection of the extrapolation points is illustrated in Figure 3.3b. The blue markers shown the discrete edge velocity which is obtained by an inviscid solver. The four discrete positions shown by the red markers are used for the extrapolation of the stagnation point.



(a) Edge velocity distribution

(b) Relevant points for extrapolation

Figure 3.3.: Extrapolation of the stagnation point.

The two positions before the stagnation location ($x^{(n-1)}$ and $x^{(n)}$) and the two positions after the stagnation location ($x^{(n+1)}$ and $x^{(n+2)}$) are used to define the two straight lines

$$U_{e,1}(x_{stag}) = U_e(x^{(n)}) + m_1\left(x_{stag} - x^{(n)}\right) \qquad \text{with} \quad m_1 = \frac{U_e(x^{(n)}) - U_e(x^{(n-1)})}{x^{(n)} - x^{(n-1)}} \tag{3.7a}$$

$$U_{e,2}(x_{stag}) = U_e(x^{(n+1)}) + m_2\left(x_{stag} - x^{(n+1)}\right) \qquad \text{with} \quad m_2 = \frac{U_e(x^{(n+2)}) - U_e(x^{(n+1)})}{x^{(n+2)} - x^{(n+1)}} \tag{3.7b}$$

which will cross at the stagnation point. The slopes of the straight lines before and after the stagnation

point, m_1 and m_2, are determined with four points marked in red in Figure (3.4). As the straight lines are linear the extrapolation is linear as well. The stagnation point is assumed at the location where the two straight lines cross which is calculated to be

$$x_{stag} = \frac{U_e(x^{(n)}) - U_e(x^{(n+1)}) - m_1 x^{(n)} + m_2 x^{(n+1)}}{m_2 - m_1}. \tag{3.8}$$

As depicted in Figure (3.4) the lines do not cross at $U_e = 0$ and thus the velocity is manually set to be zero which is the correct value for the stagnation point.

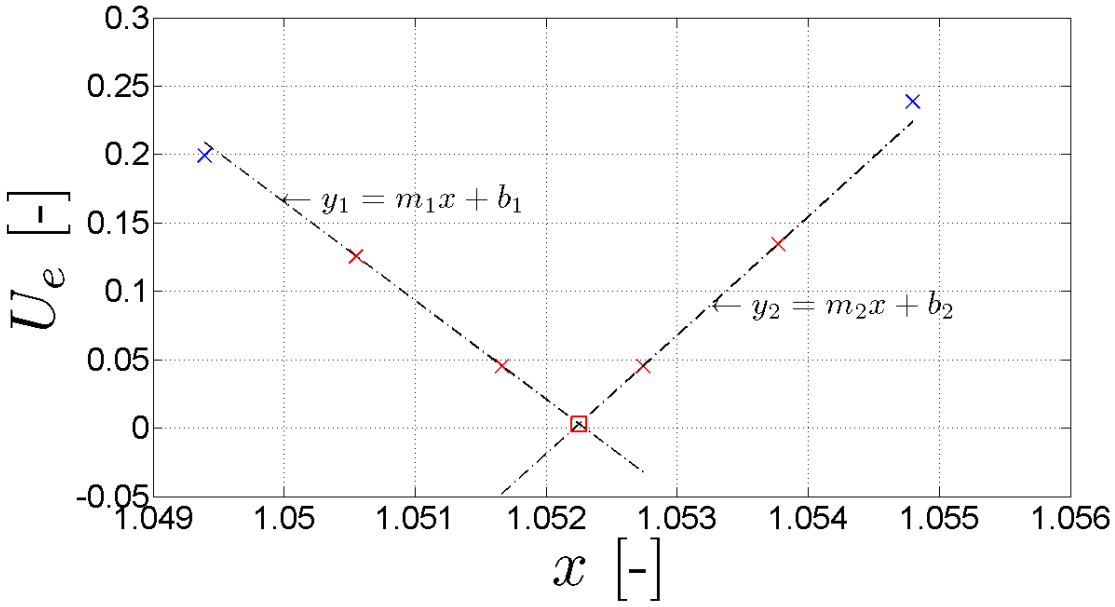


Figure 3.4.: The arc length of the stagnation point is located where the two straight lines cross.

The accuracy could be increased if more points before and after the stagnation point are included and a more complicated formula instead of (3.7) can be derived. However, a relatively fine grid spacing in streamwise direction is provided resulting in a small error and thus the linear extrapolation is used to find the stagnation point. If the pressure distribution is obtained by experimental data the spacing between the sensors could be quite large and a more accurate calculation instead of equation (3.7) should be applied.
In order to perform a complete boundary-layer calculation for an airfoil a calculation for the suction and the pressure side have to be performed each starting from the stagnation point. The first position has to be the stagnation point ($C_p = 1$) from where the arc length is computed with equation (3.6). Therefore, the first grid point belongs to the stagnation point and n is counted till the trailing edge where the total number of grid points in streamwise direction N is related to. Note that N could be different for the suction and the pressure side depending on the location of the stagnation point and the discretization used for each side of the airfoil. The subroutine to find the stagnation point is given in `linetrap.m`.

3.1.3 Differentiation Operator

Throughout the boundary-layer calculation several derivatives have to be computed. In the following an elegant method is introduced that uses matrix multiplications to compute the derivatives. In the case of

the gradient of the edge velocity $U_e(x)$ that depends on the streamwise coordinate x, the differentiation becomes

$$\frac{\partial U_e(x)}{\partial x} = \boldsymbol{D_1} U_e \tag{3.9}$$

where the subscript $_1$ stands for a single partial derivative. To build such a differentiation operator the derivatives are approximates by using Tayler series expansion.
At the first grid point a forward differencing scheme has to be applied because no previous grid points exist. In the following, it is assumed that x_1 is the position of the first grid point succeeded by x_2 and x_3 respectively. To be as general as possible, the distance between each point is arbitrarily chosen (non-uniform grid) and the resulting value at the x-positions will be $f(x)$. A sketch for the forward differencing scheme is shown in Figure 3.5a. For example, in equation (3.9) the function would be $f(x) = U_e(x)$. The distance between the grid points is given by $d_{1,1} = x_2 - x_1$ and $d_{1,2} = x_3 - x_1$ and thus the Taylor series becomes

$$f(x_2) = f(x_1) + d_{1,1} f'(x_1) + \frac{d_{1,1}^2}{2} f''(x_1) + \mathcal{O}(d_{1,1}^3) \quad \text{with} \quad d_{1,1} = x_2 - x_1 \tag{3.10a}$$

$$f(x_3) = f(x_1) + d_{1,2} f'(x_1) + \frac{d_{1,2}^2}{2} f''(x_1) + \mathcal{O}(d_{1,2}^3) \quad \text{with} \quad d_{1,2} = x_3 - x_1. \tag{3.10b}$$

By neglecting the terms of third order or higher order, the system (3.10) can be solved for $f'(x_1)$ and $f''(x_1)$ because all other terms are known. It follows

$$f'(x_1) = -\frac{d_{1,1} + d_{1,2}}{d_{1,1} d_{1,2}} f(x_1) - \frac{d_{1,2}}{d_{1,1}\left(d_{1,1} - d_{1,2}\right)} f(x_2) + \frac{d_{1,1}}{d_{1,2}\left(d_{1,1} - d_{1,2}\right)} f(x_3) \tag{3.11}$$

$$f''(x_1) = \frac{2}{d_{1,1} d_{1,2}} f(x_1) + \frac{2}{d_{1,1}\left(d_{1,1} - d_{1,2}\right)} f(x_2) - \frac{2}{d_{1,2}\left(d_{1,1} - d_{1,2}\right)} f(x_3). \tag{3.12}$$

In the case that derivatives at a midpoint have to be evaluated the accuracy is increased by taking centered differences. The corresponding sketch is shown in Figure 3.5b. Again, a Taylor series expansion is introduced with the position at the center x_n. It follows

$$f(x_{n-1}) = f(x_n) + d_{n,1} f'(x_n) + \frac{d_{n,1}^2}{2} f''(x_n) + \mathcal{O}(d_{n,1}^3) \quad \text{with} \quad d_{n,1} = x_{n-1} - x_n \tag{3.13a}$$

$$f(x_{n+1}) = f(x_n) + d_{n,2} f'(x_n) + \frac{d_{n,2}^2}{2} f''(x_n) + \mathcal{O}(d_{n,2}^3) \quad \text{with} \quad d_{n,2} = x_{n+1} - x_n. \tag{3.13b}$$

By neglecting the terms of third order or higher order, the system (3.13) can be solved for $f'(x_n)$ and $f''(x_n)$ because all other terms are known. It follows

$$f'(x_n) = -\frac{d_{n,2}}{d_{n,1}\left(d_{n,1} - d_{n,2}\right)} f(x_{n-1}) - \frac{d_{n,1} + d_{n,2}}{d_{n,1} d_{n,2}} f(x_n) + \frac{d_{n,1}}{d_{n,2}\left(d_{n,1} - d_{n,2}\right)} f(x_{n+1}) \tag{3.14}$$

$$f''(x_1) = \frac{2}{d_{n,1}\left(d_{n,1} - d_{n,2}\right)} f(x_{n-1}) + \frac{2}{d_{n,1} d_{n,2}} f(x_n) - \frac{2}{d_{n,2}\left(d_{n,1} - d_{n,2}\right)} f(x_{n+1}). \tag{3.15}$$

And finally with the same approach a backward differencing scheme for the last position can be derived which is shown in Figure 3.5c. Assume that the last position is denoted by x_N the Taylor series expansion becomes

$$f(x_{N-1}) = f(x_N) + d_{N,1} f'(x_N) + \frac{d_{N,1}^2}{2} f''(x_N) + \mathcal{O}(d_{N,1}^3) \quad \text{with} \quad d_{N,1} = x_{N-1} - x_N \tag{3.16a}$$

$$f(x_{N-2}) = f(x_N) + d_{N,2} f'(x_N) + \frac{d_{N,2}^2}{2} f''(x_N) + \mathcal{O}(d_{N,2}^3) \quad \text{with} \quad d_{N,2} = x_{N-2} - x_N. \tag{3.16b}$$

By neglecting the terms of third order or higher order, the system (3.16) can be solved for $f'(x_N)$ and $f''(x_N)$ because all other terms are known. It follows

$$f'(x_N) = \frac{d_{N,1}}{d_{N,2}\left(d_{N,1} - d_{N,2}\right)} f(x_{N-2}) - \frac{d_{N,2}}{d_{N,1}\left(d_{N,1} - d_{N,2}\right)} f(x_{N-1}) - \frac{d_{N,1} + d_{N,2}}{d_{N,1} d_{N,2}} f(x_N) \tag{3.17}$$

$$f''(x_N) = -\frac{2}{d_{N,2}\left(d_{N,1} - d_{N,2}\right)} f(x_{N-2}) + \frac{2}{d_{N,1}\left(d_{N,1} - d_{N,2}\right)} f(x_{N-1}) + \frac{2}{d_{N,1} d_{N,2}} f(x_N). \tag{3.18}$$

Finally, the differencing schemes (3.11), (3.14) and (3.17) are used to build the operator $\boldsymbol{D_1}$ which can be applied to take a single derivative with respect to x which reads

$$\underbrace{\begin{pmatrix} f'(x_1) \\ \vdots \\ f'(x_n) \\ \vdots \\ f'(x_N) \end{pmatrix}}_{f'(\boldsymbol{x})} = \underbrace{\begin{pmatrix} D_{1,1} & D_{1,2} & D_{1,3} & & & & \\ \ddots & \ddots & \ddots & & & & \\ & D_{n,n-1} & D_{n,n} & D_{n,n+1} & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & D_{N,N-2} & D_{N,N-1} & D_{N,N} \end{pmatrix}}_{\boldsymbol{D_1}} \underbrace{\begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \\ \vdots \\ f(x_N) \end{pmatrix}}_{f(\boldsymbol{x})} \tag{3.19}$$

with the coefficients from (3.11), (3.14) and (3.17)

$$D_{1,1} = -\frac{d_{1,1} + d_{1,2}}{d_{1,1} d_{1,2}} \qquad D_{1,2} = -\frac{d_{1,2}}{d_{1,1}\left(d_{1,1} - d_{1,2}\right)} \qquad D_{1,3} = \frac{d_{1,1}}{d_{1,2}\left(d_{1,1} - d_{1,2}\right)} \tag{3.20a}$$

$$D_{n,n-1} = -\frac{d_{n,2}}{d_{n,1}\left(d_{n,1} - d_{n,2}\right)} \qquad D_{n,n} = -\frac{d_{n,1} + d_{n,2}}{d_{n,1} d_{n,2}} \qquad D_{n,n+1} = \frac{d_{n,1}}{d_{n,2}\left(d_{n,1} - d_{n,2}\right)} \tag{3.20b}$$

$$D_{N,N-2} = \frac{d_{N,1}}{d_{N,2}\left(d_{N,1} - d_{N,2}\right)} \qquad D_{N,N-1} = -\frac{d_{N,2}}{d_{N,1}\left(d_{N,1} - d_{N,2}\right)} \qquad D_{N,N} = -\frac{d_{N,1} + d_{N,2}}{d_{N,1} d_{N,2}}. \tag{3.20c}$$

In the similar way the matrix for $\boldsymbol{D_2}$ can be computed. In the code the differentiation matrices are implemented in the subroutine `diffop.m`.

Another scheme should be mentioned here which is used in the *MATLAB* code to calculated the derivatives of the velocities u and v in streamwise direction. So far, in the backward differencing scheme the distances between the grid point were given directly. In the code only the difference between the current and the previous gridpoint is calculated explicitly as shown in Figure 3.5d. In order to derive the backward differencing scheme as in equation (3.16) to compute the partial difference (3.17) also the distance between the last two grid points is needed. In the numerical code the current distance of the

previous and current grid point is calculated and the distance of the last computation position is simply passed. The difference between this backward scheme and the one shown above can be compared in Figures 3.5c and 3.5d. Therefore, the equivalent Taylor series reads

$$f(x_{n-1}) = f(x_n) - \Delta x_1 f'(x_n) + \frac{\Delta x_1^2}{2} f''(x_n) + \mathcal{O}(\Delta x_1^3) \tag{3.21a}$$

$$f(x_{n-2}) = f(x_n) - (\Delta x_1 + \Delta x_2) f'(x_n) + \frac{(\Delta x_1 + \Delta x_2)^2}{2} f''(x_n) + \mathcal{O}((\Delta x_1 + \Delta x_2)^3) \tag{3.21b}$$

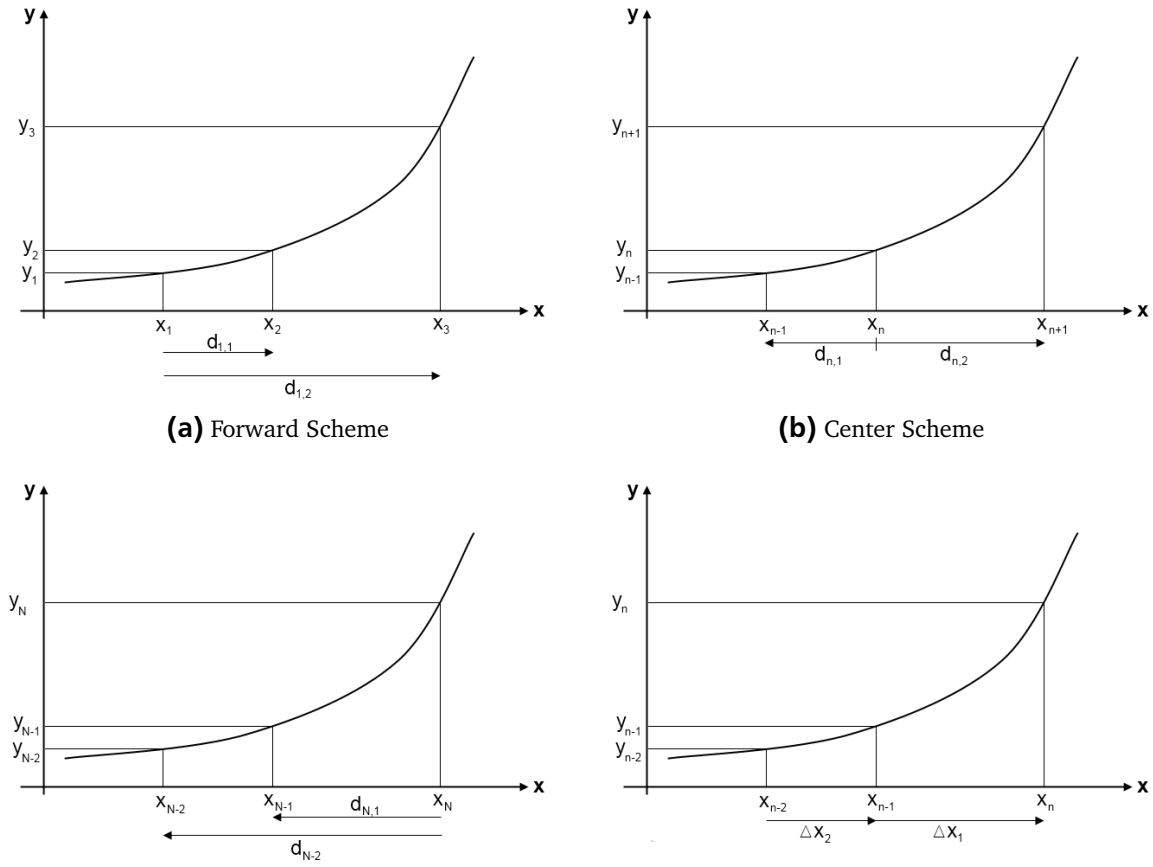
with

$$\Delta x_1 = x_n - x_{n-1} \quad \text{and} \quad \Delta x_2 = x_{n-1} - x_{n-2}. \tag{3.21c}$$

During the calculation process only the first derivative is needed which results in

$$f'(x_n) = \frac{\Delta x_1}{\Delta x_2 (\Delta x_1 + \Delta x_2)} f(x_{n-2}) - \frac{\Delta x_1 + \Delta x_2}{\Delta x_1 \Delta x_2} f(x_{n-1}) - \frac{2\Delta x_1 + \Delta x_2}{\Delta x_1 (\Delta x_1 + \Delta x_2)} f(x_n). \tag{3.22}$$

In the code that algorithm is implemented in the function `backdiffcoef.m`.



(a) Forward Scheme

(b) Center Scheme

(c) Backward Scheme

(d) Backward Scheme in x-direction

Figure 3.5.: Sketches for the derivation of the differencing schemes.

3.1.4 Tridiagonal Matrix Solver - Thomas Algorithm

The Thomas algorithm described in the following is used to solve a linear system of equations that consists of a sparse matrix $\boldsymbol{A}$ including 3x3 sub-matrices in block tridiagonal form [10]. The system of equation that is solved has the following form

$$\underbrace{\begin{pmatrix} \boldsymbol{A}_{(0)}^{(i)} & \boldsymbol{C}_{(0)}^{(i)} & & & & & & \\ \boldsymbol{B}_{(1)}^{(i)} & \boldsymbol{A}_{(1)}^{(i)} & \boldsymbol{C}_{(1)}^{(i)} & & & & & \\ & \ddots & \ddots & \ddots & & & & \\ & & \boldsymbol{B}_{(j)}^{(i)} & \boldsymbol{A}_{(j)}^{(i)} & \boldsymbol{C}_{(j)}^{(i)} & & & \\ & & & \ddots & \ddots & \ddots & & \\ & & & & \boldsymbol{B}_{(J-1)}^{(i)} & \boldsymbol{A}_{(J-1)}^{(i)} & \boldsymbol{C}_{(J-1)}^{(i)} \\ & & & & & \boldsymbol{B}_{(J)}^{(i)} & \boldsymbol{A}_{(J)}^{(i)} \end{pmatrix}}_{\boldsymbol{A}} \underbrace{\begin{pmatrix} \vec{\delta}_{(0)}^{(i)} \\ \vec{\delta}_{(1)}^{(i)} \\ \vdots \\ \vec{\delta}_{(j)}^{(i)} \\ \vdots \\ \vec{\delta}_{(J-1)}^{(i)} \\ \vec{\delta}_{(J)}^{(i)} \end{pmatrix}}_{\boldsymbol{\delta}} = \underbrace{\begin{pmatrix} \vec{r}_{(0)}^{(i)} \\ \vec{r}_{(1)}^{(i)} \\ \vdots \\ \vec{r}_{(j)}^{(i)} \\ \vdots \\ \vec{r}_{(J-1)}^{(i)} \\ \vec{r}_{(J)}^{(i)} \end{pmatrix}}_{\boldsymbol{r}}. \tag{3.23}$$

The block elimination method involves a forward and a backward sweep. In the forward sweep the 3x3 matrix $\boldsymbol{\Delta}_{(j)}$ and the vector $\vec{w}_{(j)} = \left((w_1)_{(j)}, (w_2)_{(j)}, (w_3)_{(j)}\right)^T$ for $j = 0 \leq j \leq J$ are calculated such that

$$\boldsymbol{\Delta}_{(0)} = \boldsymbol{A}_{(0)} \tag{3.24a}$$

$$\boldsymbol{\Gamma}_{(j)}\boldsymbol{\Delta}_{(j-1)} = \boldsymbol{B}_{(j)} \quad \text{for} \quad j = 1, 2, \ldots, J \tag{3.24b}$$

$$\boldsymbol{\Delta}_{(j)} = \boldsymbol{A}_{(j)} - \boldsymbol{\Gamma}_{(j)}\boldsymbol{C}_{(j-1)} \quad \text{for} \quad j = 1, 2, \ldots, J \tag{3.24c}$$

and

$$\vec{w}_{(0)} = \vec{r}_{(0)} \tag{3.24d}$$

$$\vec{w}_{(j)} = \vec{r}_{(j)} - \boldsymbol{\Gamma}_{(j)}\vec{w}_{(j-1)} \quad \text{for} \quad j = 1, 2, \ldots, J. \tag{3.24e}$$

Note that $\boldsymbol{\Gamma}_{(j)}$ has the same structure as $\boldsymbol{B}_{(j)}$ which can be written in general form as

$$\boldsymbol{\Gamma}_{(j)} = \begin{pmatrix} (\gamma_{11})_{(j)} & (\gamma_{12})_{(j)} & (\gamma_{13})_{(j)} \\ (\gamma_{21})_{(j)} & (\gamma_{22})_{(j)} & (\gamma_{23})_{(j)} \\ 0 & 0 & 0 \end{pmatrix} \tag{3.25a}$$

whereas the third row of $\boldsymbol{\Delta}_{(j)}$ is determined by the the matrix $\boldsymbol{A}_{(j)}$ such that

$$\boldsymbol{\Delta}_{(j)} = \begin{pmatrix} (\alpha_{11})_{(j)} & (\alpha_{12})_{(j)} & (\alpha_{13})_{(j)} \\ (\alpha_{21})_{(j)} & (\alpha_{22})_{(j)} & (\alpha_{23})_{(j)} \\ 0 & -1 & -\frac{h_{(j+1)}}{2} \end{pmatrix}. \tag{3.25b}$$

The structure which is given here represents the structure of the matrices that are derived later on in section 3.4.2. By using the definition (3.24a) the coefficients in (3.25b) can be determined and by taking the inverse of $\boldsymbol{\Delta}_{(0)}$ and applying the definition (3.24b) the coefficients for $\boldsymbol{\Gamma}_{(1)}$ can be found as

$$(\gamma_{11})_{(1)} = -1 \qquad (\gamma_{12})_{(1)} = -\frac{1}{2}h_{(1)} \qquad (\gamma_{13})_{(1)} = 0 \tag{3.26a}$$

$$(\gamma_{21})_{(1)} = (s_4)_{(1)} \qquad (\gamma_{23})_{(1)} = -2\frac{(s_2)_{(1)}}{h_{(1)}} \qquad (\gamma_{22})_{(1)} = (s_6)_{(1)} + (\gamma_{23})_{(1)}. \tag{3.26b}$$

The variables s_1 till s_6 are derived in section 3.4.2. After that, $\mathbf{\Delta}_{(1)}$ can be calculated using definition (3.24c) which is used again to determine $\mathbf{\Gamma}_{(2)}$. The process goes on till all matrices $\mathbf{\Delta}_{(j)}$ for $j = 0 \leq j \leq J-1$ and $\mathbf{\Gamma}_{(j)}$ for $j = 1 \leq j \leq J$ are determined. The coefficients for each matrix are found as

$$(\alpha_{11})_{(j)} = 1 \qquad (\alpha_{12})_{(j)} = -\frac{h_{(j)}}{2} - (\gamma_{13})_{(j)} \qquad (\alpha_{13})_{(j)} = \frac{h_{(j)}}{2}(\gamma_{13})_{(j)} \tag{3.27a}$$

$$(\alpha_{21})_{(j)} = (s_3)_{(j)} \qquad (\alpha_{22})_{(j)} = (s_5)_{(j)} - (\gamma_{23})_{(j)} \qquad (\alpha_{23})_{(j)} = (s_1)_{(j)} + \frac{h_{(j)}}{2}(\gamma_{23})_{(j)} \tag{3.27b}$$

and

$$\begin{aligned}
(\gamma_{11})_{(j)} &= \frac{(\alpha_{23})_{(j-1)} + \frac{h_{(j)}}{2}\left[\left(\frac{h_{(j)}}{2}\right)(\alpha_{21})_{(j-1)} - (\alpha_{22})_{(j-1)}\right]}{DET_0} \\
(\gamma_{12})_{(j)} &= -\frac{\left(\frac{h_{(j)}}{2}\right)^2 + (\gamma_{11})_{(j)}\left[\frac{h_{(j)}}{2}(\alpha_{12})_{(j-1)} - (\alpha_{13})_{(j-1)}\right]}{DET_0} \\
(\gamma_{13})_{(j)} &= \frac{\left[(\gamma_{11})_{(j)}(\alpha_{13})_{(j-1)} + (\gamma_{12})_{(j)}(\alpha_{23})_{(j-1)}\right]}{\frac{h_{(j)}}{2}} \\
(\gamma_{21})_{(j)} &= \frac{\left[(s_2)_{(j)}(\alpha_{21})_{(j-1)} - (s_4)_{(j)}(\alpha_{23})_{(j-1)} + \frac{h_{(j)}}{2}\left((s_4)_{(j)}(\alpha_{22})_{(j-1)} - (s_6)_{(j)}(\alpha_{21})_{(j-1)}\right)\right]}{DET_0} \\
(\gamma_{22})_{(j)} &= \frac{\frac{h_{(j)}}{2}(s_6)_{(j)} - (s_2)_{(j)} + (\gamma_{21})_{(j)}\left[(\alpha_{13})_{(j-1)} - \frac{h_{(j)}}{2}(\alpha_{12})_{(j-1)}\right]}{DET_1} \\
(\gamma_{22})_{(j)} &= (\gamma_{21})_{(j)}(\alpha_{12})_{(j-1)} + (\gamma_{22})_{(j)}(\alpha_{22})_{(j-1)} - (s_6)_{(j)}
\end{aligned} \tag{3.27c}$$

with the determinants

$$\begin{aligned}
DET_0 =& (\alpha_{13})_{(j-1)}(\alpha_{21})_{(j-1)} - (\alpha_{23})_{(j-1)}(\alpha_{11})_{(j-1)} - \\
&- \frac{h_{(j)}}{2}\left((\alpha_{12})_{(j-1)}(\alpha_{21})_{(j-1)} - (\alpha_{22})_{(j-1)}(\alpha_{11})_{(j-1)}\right) \\
DET_1 =& \frac{h_{(j)}}{2}(\alpha_{22})_{(j-1)} - (\alpha_{23})_{(j-1)}.
\end{aligned} \tag{3.27d}$$

After calculating all the values for $\mathbf{\Delta}_{(j)}$ and $\mathbf{\Gamma}_{(j)}$ definitions (3.24d) and (3.24e) are used to determine the values for the vectors $\vec{w}_{(j)}$.
In the backward sweep the calculated values for $\mathbf{\Delta}_{(j)}$ and $\vec{w}_{(j)}$ are used to get $\delta f_{(j)}$, $\delta u_{(j)}$ and $\delta \nu_{(j)}$ from the Newton method. Those are computed from the boundary-layer edge to the wall by

$$\mathbf{\Delta}_{(J)}\vec{\delta}_{(J)} = \vec{w}_{(J)} \tag{3.28a}$$

$$\mathbf{\Delta}_{(j)}\vec{\delta}_{(j)} = \vec{w}_{(j)} - \boldsymbol{C}_{(j)}\vec{\delta}_{(j+1)} \quad \text{for} \quad j = J-1, J-2, \ldots, 0. \tag{3.28b}$$

Similarly to the forward sweep, the iterates of the Newton method at the boundary-layer edge are found to be

$$\delta u_{(J)}^{(i)} = (w_3)_{(J)} \tag{3.29a}$$

$$\delta v_{(J)}^{(i)} = \frac{e_2(\alpha_{11})_{(J)} - e_1(\alpha_{21})_{(J)}}{(\alpha_{23})_{(J)}(\alpha_{11})_{(J)} - (\alpha_{13})_{(J)}(\alpha_{21})_{(J)}} \tag{3.29b}$$

$$\delta f_{(J)}^{(i)} = \frac{e_1 - (\alpha_{13})_{(J)}\delta v_{(J)}^{(i)}}{(\alpha_{11})_{(J)}} \tag{3.29c}$$

and all other positions are given as

$$\delta v_{(j)}^{(i)} = \frac{(\alpha_{11})_{(j)}\left[(w_2)_{(j)} + e_3(\alpha_{22})_{(j)}\right] - (\alpha_{21})_{(j)}(w_1)_{(j)} - e_3(\alpha_{21})_{(j)}(\alpha_{12})_{(j)}}{DET_2} \tag{3.30a}$$

$$\delta u_{(j)}^{(i)} = -\frac{h_{(j+1)}}{2}\delta v_{(j)}^{(i)} - e_3 \tag{3.30b}$$

$$\delta f_{(j)}^{(i)} = \frac{(w_1)_{(j)} - (\alpha_{12})_{(j)}\delta u_{(j)}^{(i)} - (\alpha_{13})_{(j)}\delta v_{(j)}^{(i)}}{(\alpha_{11})_{(j)}} \tag{3.30c}$$

with

$$e_1 = (w_1)_{(J)} - (\alpha_{12})_{(J)}\delta u_{(J)}^{(i)} \tag{3.31a}$$

$$e_2 = (w_2)_{(J)} - (\alpha_{22})_{(J)}\delta u_{(J)}^{(i)} \tag{3.31b}$$

$$e_3 = (w_3)_{(j)} - \delta u_{(j+1)}^{(i)} + \frac{h_{(j+1)}}{2}\delta v_{(j+1)}^{(i)} \tag{3.31c}$$

$$\begin{aligned} DET_2 = &\frac{h_{(j+1)}}{2}(\alpha_{21})_{(j)}(\alpha_{12})_{(j)} - (\alpha_{21})_{(j)}(\alpha_{13})_{(j)} - \\ &- \frac{h_{(j+1)}}{2}(\alpha_{22})_{(j)}(\alpha_{11})_{(j)} + (\alpha_{23})_{(j)}(\alpha_{11})_{(j)}. \end{aligned} \tag{3.31d}$$

The routine which is described from equations (3.24) to (3.31) is build into *MATLAB* when solving systems of equations that have a block tridiagonal form. More details about the Thomas algorithm can be found in [10].

3.2 Solutions of the Integral Boundary-Layer Equation

The integral boundary-layer equation which is derived in section 2.3 can be solved by different assumptions as outlined in section 2.4. In the following, the second Walz method from section 2.4.3 and Thwaites method from section 2.4.4 are implemented and the characteristic solutions are shown.

3.2.1 Walz Method

The implementation is straight forward as the relevant equations are given explicitly in section 2.4.3. The streamwise grid points are taken from the inviscid flow solver. The calculation has to start at the

stagnation point which correct position is determined by the procedure described in section 3.1.2. After that the integral given in equation (2.60) is numerically discretized as

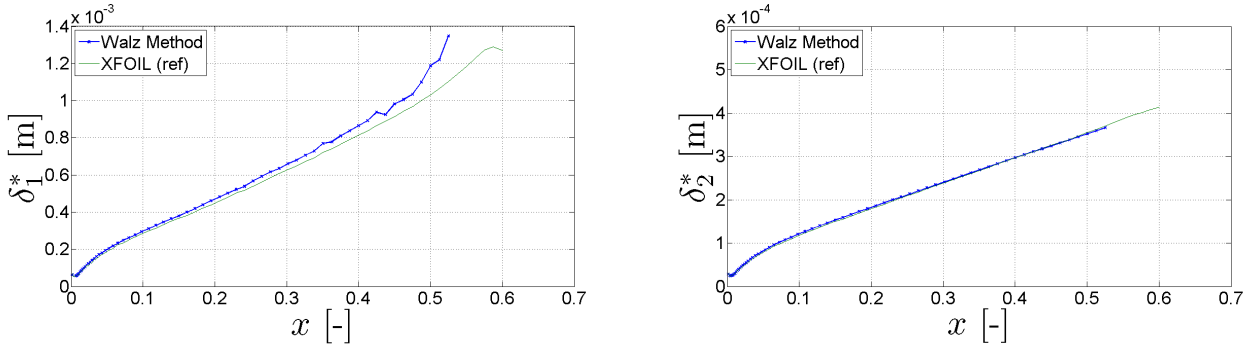
$$Z^{(n+1)} = \left(\frac{U_e^{(n)}}{U_e^{(n+1)}}\right)^b Z^{(n)} + \frac{a}{1+a}\frac{1-\left(U_e^{(n)}/U_e^{(n+1)}\right)^{b+1}}{1-\left(U_e^{(n)}/U_e^{(n+1)}\right)}\left(x^{(n+1)}-x^{(n)}\right) \tag{3.32}$$

where the coefficient a and b are found in equation (2.58). The calculation starts at $Z^{(0)} = 0$. After having obtained $Z^{(n+1)}$ the second parameter Γ is found by equation (2.54) which is numerically given as

$$\Gamma^{(n+1)} = \frac{Z^{(n+1)}}{U_e^{(n+1)}}\frac{U_e^{(n+1)}-U_e^{(n)}}{x^{(n+1)}-x^{(n)}}. \tag{3.33}$$

The calculation stops at the separation point which is reached when $\Gamma = -0.0681$. Note that the coefficients a and b vary for different Γ.

The results for a sample calculation of the *MW-airfoil* at an angle of attack $\alpha = 3.3°$ and a Reynolds number of $Re = 2\,351\,842$ for the suction side is shown in Figure 3.6. The calculation especially solves for Z which can be directly related to the momentum thickness as shown in Figure 3.6b. An implication of that is the very good agreement with the reference data given by *XFOIL*. Slight discrepancies are found because the *XFOIL* results show the converged solution which includes viscous-inviscid interaction and thus the results are based on a slightly different pressure distribution. In contrast to the momentum thickness the displacement thickness is indirectly calculated. The displacement thickness is determined by the shape factor H_{12} that relates the momentum and displacement thickness over the ratio. The shape factor that corresponds to the calculated Γ at each position is computed by a Falkner-Skan solver that is described in section 3.3. With those results the shape factor is calculated for discrete values of the Hartree parameter which then can be related to Γ. In the end the correct shape factor is interpolated and thus the wavy trend of the displacement thickness in Figure 3.6a can be explained. The separation point is computed too early which could be explained by the slightly different pressure distribution.



(a) Displacement thickness

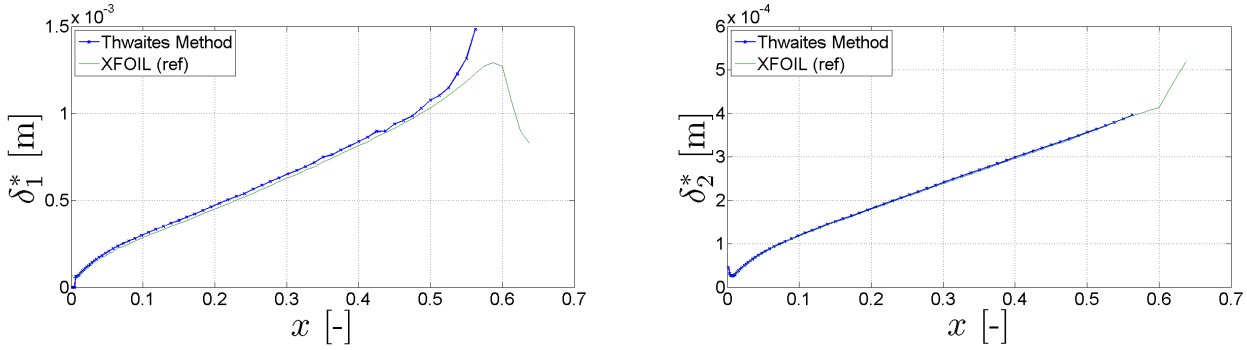
(b) Momentum thickness

Figure 3.6.: Boundary-layer characteristics of the suction side for the Walz method (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

3.2.2 Thwaites Method

Like the method of Walz the implementation for Thwaites method is also straight forward. The numerical grid is taken from the potential flow solver and the stagnation point is found by the procedure described

in section 3.1.2. The calculation starts at the stagnation point from where the integral of equation (2.67) is solved for an initial value of the displacement thickness given by equation (2.68). After having computed the displacement thickness, the dimensionless pressure gradient λ, which is given in equation (2.45), is found. The value for λ is then correlated to l and the shape factor H_{12} which are defined in equations (2.69) and (2.70), respectively. Note that the range for λ is restricted and that outside the range the solution cannot be given.
In Figure 3.7 the boundary-layer characteristics for the suction side are depicted. The flow is computed for the *MW-airfoil* at an angle of attack of $\alpha = 3.3°$ for a Reynolds number of $Re = 2\,351\,842$. The computation is directly operated to get the momentum thickness which is shown in Figure 3.7b. The agreement is very good when compared to the reference data from *XFOIL*. Further the separation point is found later than for the Walz method. The separation point is referred to the location where the dimensionless pressure gradient λ is out of range. As illustrated in Figure 3.7a the displacement thickness matches not that good with the reference data but the agreement is better than for the calculation of Walz method. Note that at the beginning of the calculation the computed λ is out of range. As the angle of attack is positive, the stagnation point is at the lower side of the airfoil and thus the flow has to go around the leading edge of the airfoil. Those positions are out of range and the value is automatically set to zero for those positions as shown in Figure 3.7a.



(a) Displacement thickness

(b) Momentum thickness

Figure 3.7.: Boundary-layer characteristics of the suction side for the Thwaites method (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

3.3 Solution Method for Similarity Flows

In the following section the similarity solution with a numerical ordinary differential equation solver is presented. The similarity solution is important as it can be used as an initial profile for the solution of airfoil flows which are discussed in the next section. A Newton method is implemented to find the similarity profile in dependence of the Hartree parameter β_H (2.86) and a given grid spacing. The first grid point in streamwise direction ξ_{start} is defined by the first position x from which the calculation is started. The transformed wall-normal variable (2.83)

$$\eta \equiv \eta_\beta = y\sqrt{\frac{m+1}{2}Re\frac{U_e(x)}{x}}$$

and the Falkner-Skan equation (2.87)

$$f_{\eta\eta\eta} + f f_{\eta\eta} + \beta_H\left(1 - f_\eta^2\right) = 0$$

with the boundary conditions (2.77)

$$\begin{aligned} \eta = 0: &\qquad f = 0 \quad \text{and} \quad f_\eta = 0 \\ \eta \to \infty: &\qquad f_\eta = 1 \end{aligned}$$

is solved. The parameter η_β will be denoted η in the following for simplicity. Note that the following procedure could also be performed for the transformed wall-normal distance (2.71) and the originally derived Falkner-Skan equation (2.78). In that case the derived equations of the next section would be slightly different but the approach would be the same.
A convenient method for solving the third order nonlinear two-point boundary value problem given by equations (2.87) and (2.77) is described by Cebeci and Keller [9]. The third order ordinary differential equation (2.87) is replaced by a system of three first-order differential equations. Each unknown (f, f_η and $f_{\eta\eta}$) is substituted with

$$f_\eta = u \tag{3.34a}$$
$$f_{\eta\eta} = u' = v \tag{3.34b}$$
$$f_{\eta\eta\eta} = v' = -f\,v - \beta_H\left(1 - u^2\right) \tag{3.34c}$$

where (3.34c) represents the Falkner-Skan equation and f, u and v are the unknowns for which must be solved for. The prime ($'$) denotes a differentiation with respect to η. In order to solve the system of equations with *MATLAB* a transformation into vector and matrix form

$$\boldsymbol{\phi}' = \boldsymbol{g}(\boldsymbol{\phi}) \quad \Leftrightarrow \quad \begin{pmatrix} f \\ u \\ v \end{pmatrix}' = \begin{pmatrix} u \\ v \\ -f\,v - \beta_H\left(1 - u^2\right) \end{pmatrix} \tag{3.35}$$

is useful. The boundary conditions transform into

$$\begin{aligned} \eta = 0: &\qquad f = 0 \quad \text{and} \quad u = 0 \\ \eta = \eta_{max}: &\qquad u = 1. \end{aligned} \tag{3.36}$$

Due to the fact that a boundary value problem has to be solved and *MATLAB* needs initial values for f, u and v. The initial value $v(0)$ has to be found solving for the boundary condition at the boundary-layer edge. The iteration condition reads

$$v(0) = s \tag{3.37}$$

and the unknowns become functions of η and s such that $\boldsymbol{\phi} = \left[f(\eta,s), u(\eta,s), v(\eta,s)\right]^T$. These have to fulfill the boundary condition at the boundary-layer edge

$$u(\eta_{max}, x) - 1 = \varphi(s) = 0. \tag{3.38}$$

A Newton method can be derived to find the correct value for s and thus solves the two-point boundary-value problem (3.35)

$$s^{i+1} = s^i - \frac{\varphi(s^i)}{\frac{d\varphi(s^i)}{ds}} = s^i - \frac{u(\eta_{max}, s^i) - 1}{\frac{\partial u(\eta_{max}, s)}{\partial s}} \quad \text{for} \quad i = 0, 1, 2, \ldots \tag{3.39}$$

where i is the number of iterations. The new unknown $\frac{\partial u(\eta_{max},s)}{\partial s}$ is found by taking the derivative of $\boldsymbol{\phi}$ with respect to s

$$f_s = F' = U \tag{3.40a}$$
$$f_{ss} = U' = V \tag{3.40b}$$
$$f_{sss} = V' = -fV - \nu F + 2u\beta_H U \tag{3.40c}$$

which is an initial value problem with the initial conditions

$$F(0) = 0, \quad U(0) = 0 \quad \text{and} \quad V(0) = 0. \tag{3.41}$$

In this case the prime ($'$) denotes a derivative with respect to s. By evaluating U at the edge of the boundary layer, the new candidate for s in equation (3.39) is obtained. That iteration process is repeated until the boundary conditions (3.36) are fulfilled. The convergence rate for Newton's method is quadratic and thus a solution is quickly found. As Newton's method does not ensure global convergence, a suitable initial guess for s has to be given. Note that the described method only works for positive wall shear stress which is found for $-0.19884 \leq \beta_H \leq 2$. The profile with $\beta_H = -0.19884$ defines the separation profile and thus can be used to define a separation point.
In Figure 3.8 the solutions for the Blasius case ($\beta_H = 0$) are shown. Note that the first derivative f_η approaches asymptotically one whereas the second derivative $f_{\eta\eta}$ approaches zero. When those derivatives are constant the solution of f rises linear. In Figure 3.8 the solution of f is scaled with 0.1 to fit all plots in one scale.

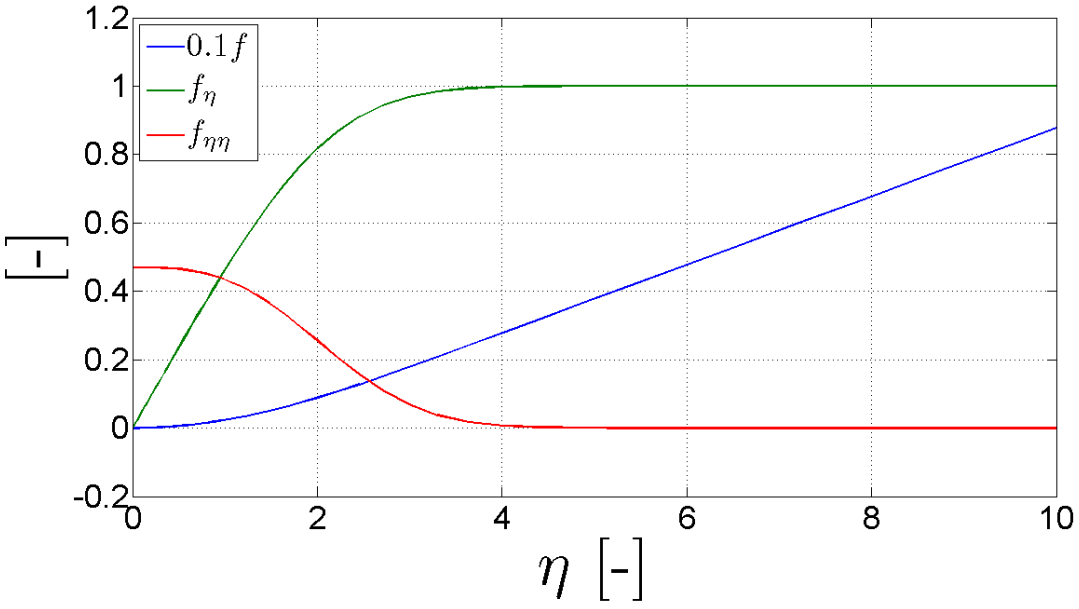


Figure 3.8.: Falkner-Skan solutions for the Blasius case ($\beta_H = 0$).

The Falkner-Skan profiles for different values of the Hartree parameter is shown in Figure 3.9. For increasing β_H the profiles become fuller which is in correspondence with theory. Therefore, the boundary-layer, the displacement and the momentum thickness decrease with increasing β_H. Note that the important cases for the separation point and the stagnation point are included. The separation point is

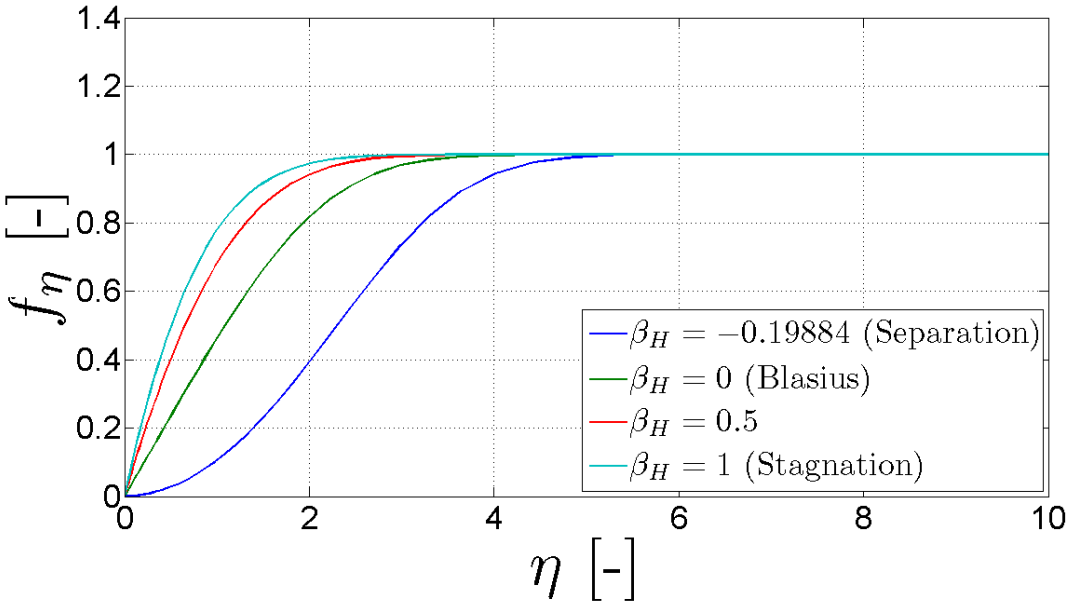


Figure 3.9.: Falkner-Skan solutions for different Hartree parameters.

characterized by a vanishing second derivative at the wall.
In order to find an exact value for η_{max} Asaithambi [3] presents an interesting approach. In that case a second iteration process is involved that iterates η_{max}. An important feature is that the initial guess for s does not have to be close to the exact solution as long as it is smaller than the exact value. Whereas Asaithambi [3] uses a secant method to iterate the solution, Zhang and Chen [29] introduced a real Newton method to solve the problem. However, the original approach to iterate only s is sufficient for a good solution and thus the numerical scheme of Cebeci and Keller [9] is used in the following.

3.4 Discretization of the Laminar Boundary-Layer Equation for Airfoil Flows

In contrast to similarity flows where the pressure gradient is constant and thus similarity profiles can be found which do not change in shape, airfoil flows do not allow such a simple solution. As a matter of fact the pressure gradient varies over the airfoil and thus the right-hand side of the equations (2.75) and (2.97) do not vanish. The result is a partial differential equation instead of an ordinary differential equations as in case of the similarity flows where the right-hand side cancels. However, the solution of the similarity flow can be used as an initial profile at the stagnation point which will be done for the Keller box discretization in subsection 3.4.2.
An additional problem is the fact that the boundary-layer calculation has to start at the stagnation point ($\xi_{start} = x_{stag}$). In general, the exact location of the stagnation point where $C_p = 1$ is not included and thus has to be found first. A method of finding that location is presented in section 3.1.2.
Another issue that has to be solved before the actual boundary-layer calculation can start involves the computation of the derivative of the edge velocity $\frac{\partial U_e}{\partial x}$ which is needed to calculate the dimensionless pressure gradient m (2.76). Furthermore, the derivatives in streamwise and wall-normal directions have to be approximated as well. The relevant derivatives are given in section 3.1.3.
Finally, the boundary-layer equations in combination with the transformations which were derived in section 2.5 are discretized and solved by applying a numerical scheme. As the boundary-layer equations are nonlinear they have to be linearized and iterated. That part is presented for two different numerical

methods in subsections 3.4.1 and 3.4.2. In subsection 3.4.1 the boundary-layer equation is solved directly whereas in subsection 3.4.2 a solution is presented which solves for the dimensionless stream function and its derivatives.

3.4.1 Discretization of the Boundary-Layer Equation using the Direct Method

After having introduced a suitable numerical grid in section 3.1.1 and having shown an efficient method to find the position of the stagnation point in subsection 3.1.2, the boundary-layer equation (2.25) derived in section 2.2 can be discretized in combination with the Falkner-Skan transformation (2.71). The continuity equation 2.16 will be used to obtain the wall-normal velocity v. For purposes of clarity the relevant equations are stated again.

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$
$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = U_e\frac{\partial U_e}{\partial x} + \frac{1}{Re}\frac{\partial^2 u}{\partial y^2}$$

and

$$\eta = y\sqrt{Re\frac{U_e}{x}}$$
$$\frac{\partial \eta}{\partial x} = \frac{1}{2}\eta\left(\frac{\partial U_e/\partial x}{U_e} - \frac{1}{x}\right)$$
$$\frac{\partial \eta}{\partial y} = \sqrt{Re\frac{U_e}{x}}.$$

General Solution Procedure

The following calculations after the leading edge are performed with the original boundary-layer equation (2.25) and the Falkner-Skan transformation (2.71) and the derivatives of η (2.73) in x- and y-direction. The discretized equation then becomes for the nth position

$$u^{(i)}\left(\frac{\partial u^{(i)}}{\partial \eta}\frac{\partial \eta}{\partial x} + \frac{\partial u^{(i)}}{\partial x}\right) + v^{(i)}\frac{\partial u^{(i)}}{\partial \eta}\frac{\partial \eta}{\partial y} = U_e\frac{\partial U_e}{\partial x} + \frac{1}{Re}\frac{\partial^2 u^{(i)}}{\partial \eta^2}\frac{\partial^2 \eta}{\partial y^2}. \tag{3.42}$$

The derivative of the edge velocity (first term on the right-hand side) can be found be using centered differences as shown in section 3.1.3 whereas the derivative of the streamwise velocity $u^{(i)}$ in x is found by using a backward differencing scheme as the value at the next position is not known yet. The derivative will be

$$\frac{\partial u^{(i)}}{\partial x} = D_{x,n}u^{(n),(i)} + D_{x,n-1}u^{(n-1)} + D_{x,n-2}u^{(n-2)}$$

where the terms $D_{x,n}$ to $D_{x,n-2}$ are the backward derivative terms in x-direction derived in section 3.1.3. The relevant subroutine that computes the backward differencing coefficients is implemented in

`backdiffcoef.m`. By sorting old terms on the right-hand side the discretized boundary-layer equation becomes

$$\left[D_{x,n}u^{(i)} + \left(u^{(i)}\frac{\partial\eta}{\partial x} + v^{(i)}\frac{\partial\eta}{\partial y}\right)\frac{\partial}{\partial\eta} - \frac{1}{Re}\left(\frac{\partial\eta}{\partial y}\right)^2\frac{\partial^2}{\partial\eta^2}\right]u^{(i)} = U_e\frac{\partial U_e}{\partial x} - \\ - u^{(i)}\left(D_{x,n-1}u^{(n-1)} + D_{x,n-2}u^{(n-2)}\right). \tag{3.43}$$

Again, the equation can be written in matrix form (3.48) in the procedure described above for the stagnation point and then solved for $u^{(i)}$. Further, the continuity equation (2.16) is used to find the wall-normal velocity component

$$v^{(i)} = v_0^{(n)} + \frac{1}{\frac{\partial\eta}{\partial y}}\int_0^{\eta_{max}}\left[-\frac{\partial u^{(i)}}{\partial\eta}\frac{\partial\eta}{\partial x} - \left(D_{x,n}u^{(n),(i)} + D_{x,n-1}u^{(n-1)} + D_{x,n-2}u^{(n-2)}\right)\right]d\eta \tag{3.44}$$

where $v_0^{(n)}$ is the blowing or suction velocity at the wall. After that the iteration starts again till the convergence criterion is fulfilled. The convergence tolerance is set to 10^{-6} and the iteration condition is the derivative of $\partial u^{(i)}/\partial\eta$ at the first grid point off the wall ($j = 1$). When the iteration condition is satisfied the derivatives of the velocity components are obtained and the calculation starts at the next streamwise position. Note that this method cannot deal with separating flows. Therefore, the calculation should stop after a maximum count for the iteration.

Treatment of the Boundary-Layer Equation at the Stagnation point

The solution at leading edge is different, as the edge velocity is zero and thus no velocity profile for the streamwise direction can be found. In order to resolve that problem the derivative of the boundary-layer equation (2.25) with respect to x is taken resulting in

$$\left(\frac{\partial u}{\partial x}\right)^2 + \underbrace{u}_{=0}\frac{\partial^2 u}{\partial x^2} + \frac{\partial v}{\partial x}\underbrace{\frac{\partial u}{\partial y}}_{=0} + v\frac{\partial}{\partial y}\frac{\partial u}{\partial x} = \left(\frac{\partial U_e}{\partial x}\right)^2 + \underbrace{U_e}_{=0}\frac{\partial^2 U_e}{\partial x^2} + \frac{1}{Re}\frac{\partial^2}{\partial y^2}\frac{\partial u}{\partial x}.$$

All terms containing the streamwise velocity u without a derivative or with a derivatives in wall-normal direction vanish as well as the edge velocity at the stagnation point. The terms containing v or derivatives in x-direction do not cancel out. In the following the equation will be solved for $\partial u/\partial x$ which then reads

$$\left(\frac{\partial u}{\partial x} + v\frac{\partial}{\partial y} - \frac{1}{Re}\frac{\partial^2}{\partial y^2}\right)\frac{\partial u}{\partial x} = \left(\frac{\partial U_e}{\partial x}\right)^2. \tag{3.45}$$

Further, at the stagnation point the limit of η for $x \to 0$ and the derivatives with respect to x and y become

$$\eta = y\sqrt{Re\frac{\partial U_e}{\partial x}} \tag{3.46a}$$

$$\frac{\partial\eta}{\partial x} = \frac{1}{2}\eta\frac{\partial^2 U_e/\partial x^2}{\partial U_e/\partial x} \tag{3.46b}$$

$$\frac{\partial\eta}{\partial y} = \sqrt{Re\frac{\partial U_e}{\partial x}}. \tag{3.46c}$$

Note that

$$\frac{\partial}{\partial y} = \frac{\partial \eta}{\partial y}\frac{\partial}{\partial \eta}$$

which is used to simplify the derivatives in (3.45) as the equation will be solved for the numerical grid that is formulated for η. By inserting (3.46c) into (3.45), adding $(\partial u/\partial x)^2$ on both sides to increase numerical stability and then dividing by $\left(\partial U_e/\partial x\right)^2$ it follows

$$\left[2\frac{\frac{\partial u}{\partial x}}{\frac{\partial U_e}{\partial x}} + \frac{\nu}{\frac{\partial U_e}{\partial x}}\frac{\partial \eta}{\partial y}\frac{\partial}{\partial \eta} - \frac{1}{Re\frac{\partial U_e}{\partial x}}\left(\frac{\partial \eta}{\partial y}\right)^2\frac{\partial^2}{\partial \eta^2}\right]\frac{\frac{\partial u}{\partial x}}{\frac{\partial U_e}{\partial x}} = 1 + \left(\frac{\frac{\partial u}{\partial x}}{\frac{\partial U_e}{\partial x}}\right)^2. \tag{3.47}$$

This equation is solved for $\frac{\partial u}{\partial x}/\frac{\partial U_e}{\partial x}$. The equation is transferred into matrix form to ensure an efficient solution in *MATLAB*. Then equation (3.47) reads

$$\boldsymbol{L}\boldsymbol{u} = \boldsymbol{b} \tag{3.48}$$

with the matrix $\boldsymbol{L}$ in tridiagonal block form. In order to efficiently build up the matrix each term can be written as a vector having the length of the total number of grid points in η-direction. After that the command `diag` is used to build a diagonal matrix which is then multiplied by the differential operator (see section 3.1.3) that is also a matrix. That form of matrix can be efficiently solved in *MATLAB* using the build-in routine of the Thomas algorithm described in section 3.1.4.
The continuity equation (2.16)

$$\frac{\frac{\partial u}{\partial x}}{\frac{\partial U_e}{\partial x}} + \frac{\frac{\partial v}{\partial y}}{\frac{\partial U_e}{\partial x}} = 0$$

is then employed to obtain the wall-normal velocity which reads

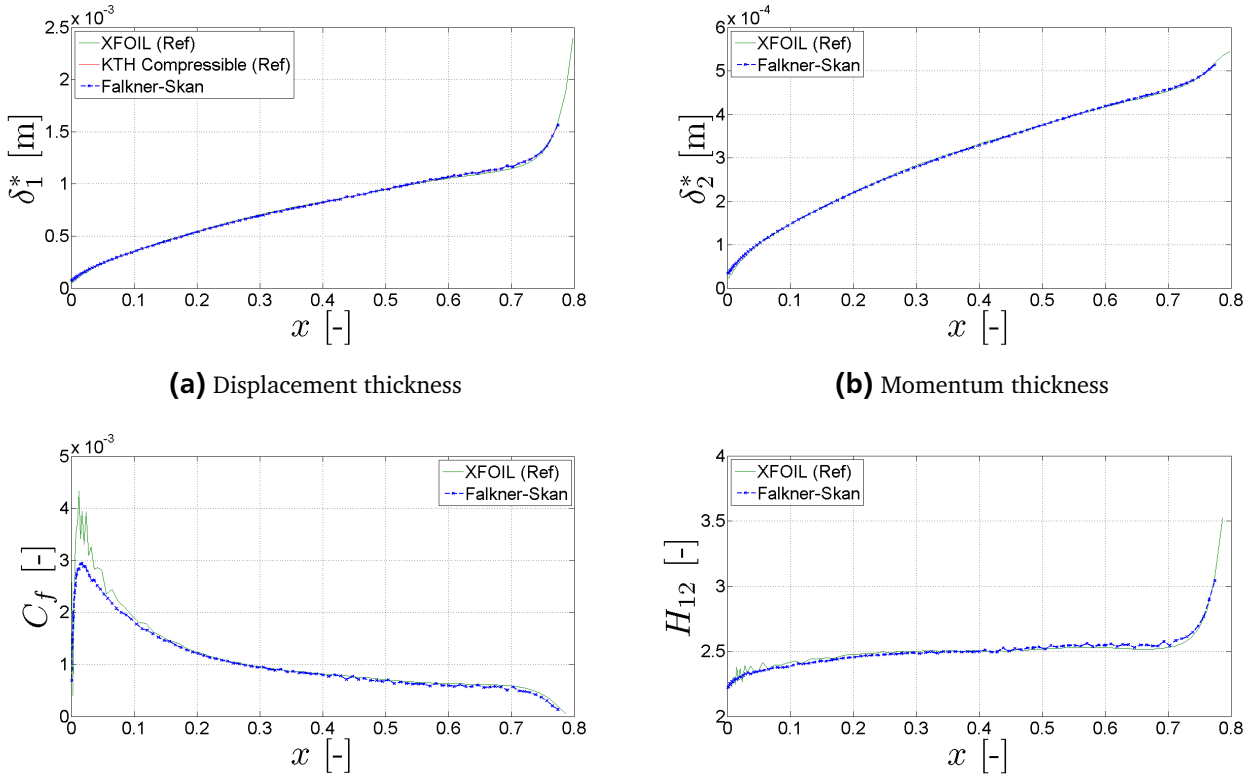
$$\frac{v}{\frac{\partial U_e}{\partial x}} = \frac{v_0}{\frac{\partial U_e}{\partial x}} + \frac{1}{\frac{\partial \eta}{\partial y}}\int_0^{\eta_{max}} -\frac{\frac{\partial u}{\partial x}}{\frac{\partial U_e}{\partial x}} d\eta. \tag{3.49}$$

In the next iteration step the wall-normal component $v/\frac{\partial U_e}{\partial x}$ is utilized as an input in (3.47) and the calculation starts at the beginning till a convergence criterion is satisfied. In the performed calculations the solution is supposed to have converged if

$$\frac{\left\{\frac{\frac{\partial u}{\partial x}}{\frac{\partial U_e}{\partial x}}\right\}_{(1)}^{(i)} - \left\{\frac{\frac{\partial u}{\partial x}}{\frac{\partial U_e}{\partial x}}\right\}_{(1)}^{(i-1)}}{\left\{\frac{\frac{\partial u}{\partial x}}{\frac{\partial U_e}{\partial x}}\right\}_{(1)}^{(i-1)}} < 10^{-6}$$

where i is the iteration counter. The relevant derivative is taken at the first grid point off the wall ($j = 1$) as changes are supposed to be maximal there. After the solution is converged the relevant velocities and derivatives in x- and y-direction are found by multiplying with $\partial U_e/\partial x$. The routine in *MATLAB* for the stagnation point is implemented in `leadingedge.m`.

The local profiles for u and v are used to compute the derivatives and the boundary-layer characteristics which are shown for pressure and section side in Figures 3.10 to 3.13. In the following the results for a calculation of the *MW-airfoil* is shown. The angle of attack for the computation is $\alpha = 3.3°$ for a Reynolds number of $Re = 2\,351\,842$ and a Mach number of $Ma = 0.01$. The Mach number is used to determine the free-stream velocity U^*_∞ in order to directly compare the results with the compressible KTH boundary-layer code. The boundary-layer code from KTH is the same as used in [14]. The boundary-layer profiles for the pressure side are depicted for a streamwise position of $x = 0.0584$ which corresponds to $x_c = 0.0594$ in airfoil coordinates. For the suction side the profiles are shown at $x = 0.0689$ which corresponds to $x_c = 0.0373$.



(a) Displacement thickness

(b) Momentum thickness

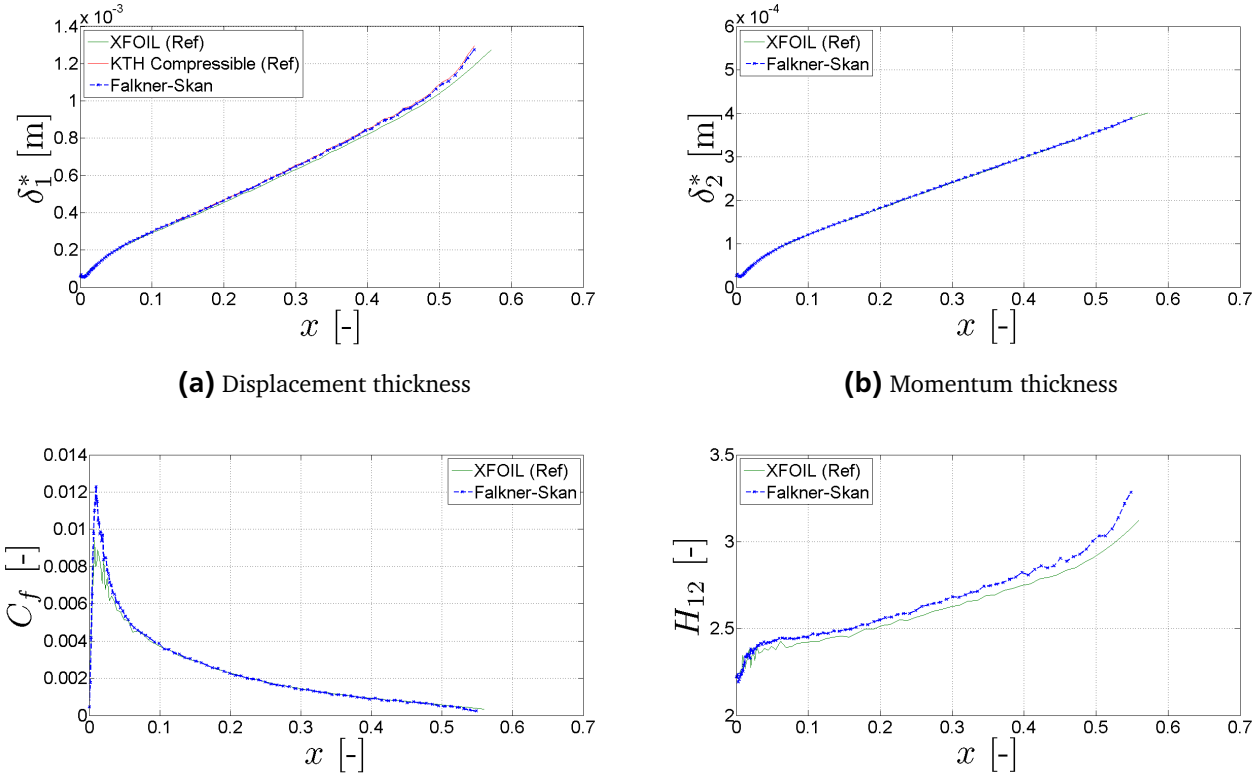
(c) Skin friction coefficient

(d) Shape factor

Figure 3.10.: Boundary-layer characteristics of the pressure side for the Falkner-Skan transformation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

The boundary-layer characteristics for the pressure side in Figure 3.10 match perfectly with the reference data of the *KTH Compressible Solver*. The value at the stagnation point in Figures 3.10a agrees with the quantities calculated by the KTH reference. Also, the last position of convergence is the same resulting in values that closely relate to the KTH reference. In contrast to that the *XFOIL* reference in Figures 3.10a to 3.10d shows the same trend but the quantities differ to a small extent. The reason for that is that *XFOIL* solves the boundary layer interactively and thus the solution belongs to a slightly different distribution of the pressure or the edge velocity. Therefore, the large discrepancy in the peak of the friction coefficient in Figure 3.10c can be explained. Furthermore, *XFOIL* uses an integral boundary-layer scheme which

is supposed to be less accurate than a direct method. However, the solution by using the Falkner-Skan tranformation demontrates very smooth curves which is clearly visible in Figures 3.10c and 3.10d when compared to the *XFOIL* solution. Further, the onset of transition is detected quite well by the solver which is symbolized by the rising shape factor in Figure 3.10d.
Quite the same results are obtained for the suction side in Figure 3.11. The profiles match perfectly with the KTH reference. The small decrease of the displacement thickness and the momentum thickness in Figures 3.11a and 3.11b at the beginning of the calculation refers to the length from the stagnation point to the leading edge. Note that for a positive angle of attack the stagnation point is at the lower side of the airfoil. From the stagnation point to the leading edge the boundary profiles get "fuller" resulting in decreasing values. After the leading edge the flow accelerates resulting in increasing values. Again the quantities in Figures 3.11a to 3.11d differ slightly compared to the *XFOIL* calculation because the results are based on a slightly different pressure distribution. Therefore, the peak in the friction coefficient in Figure 3.11c is overestimated. Note that for the pressure side the peak was underestimated. In contrast to the pressure side the onset of transition is estimated a little earlier as recognizable in Figures 3.11c and 3.11d because the skin friction coefficient approaches zero to early and the shape factor increases early. Those effects can be traced back to the slightly different pressure distribution and the different solution approach. Again, the characteristics are smoother than for the *XFOIL* reference which suggests that the scheme is numerically quite stable.



(a) Displacement thickness

(b) Momentum thickness

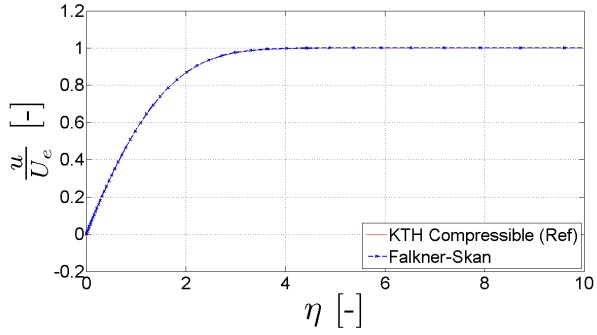
(c) Skin friction coefficient

(d) Shape factor

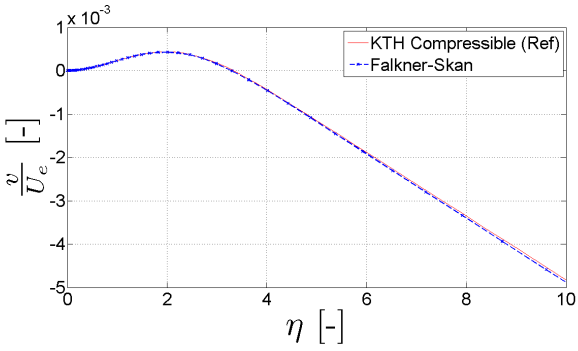
Figure 3.11.: Boundary-layer characteristics of the suction side for the Falkner-Skan transformation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

The boundary-layer profiles for the pressure side as shown in Figure 3.12 match very well with the KTH reference profiles. Note that *XFOIL* is an integral solver and thus no boundary-layer profiles can be found. The profile for the streamwise velocity u and its derivatives in wall-normal direction (Figures

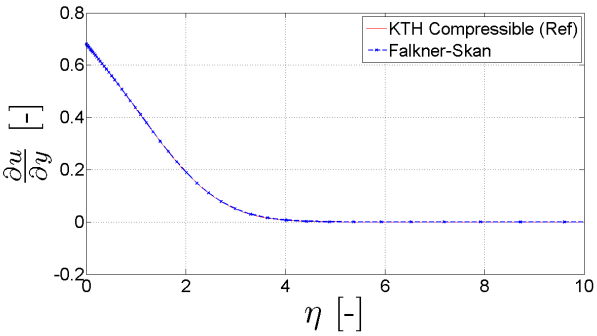
3.12a, 3.12c and 3.12e) cannot be distinguished from the reference data. A small difference for the derivative in streamwise direction in Figure 3.12g for the flow approaching the boundary-layer edge is found. The profiles computed for the v component differ slightly more. An explanation for that could be the relatively small value for those quantities which is usually three to four orders of magnitude lower. Therefore, small numerical differences become visible resulting a very small discrepancies when compared to the KTH reference. Another possible explanation is the fact that the KTH solver is written for compressible flows and that the slight change in viscosity results also in a slightly different profile.
Similar results are obtained for the suction side as illustrated in Figure 3.13. The trend and the absolute quantities match very well with the reference data from the KTH solver. The streamwise component and its derivatives agree perfectly with the reference data and no discrepancy could be found (Figures 3.13a, 3.13c, 3.13e and 3.13g). Further, also the streamwise derivative is computed on top of the reference and no difference could be found as shown in Figure 3.13g. In contrast to the pressure side, the discrepancies for the suction side are slightly more noticeable. The peak for the v components is found a little earlier and thus the derivatives in wall-normal direction also show the peak a little early. The streamwise derivative of v is in very close agreement to the reference data. The same reasons for those slight differences can be given for the suction side. The order of magnitude for the v component is much lower resulting in small numerical differences and also compressible effects have a greater impact on the v component due to the small order of magnitude.
The agreement with the reference is also very good for other positions for the suction as well as the pressure side.



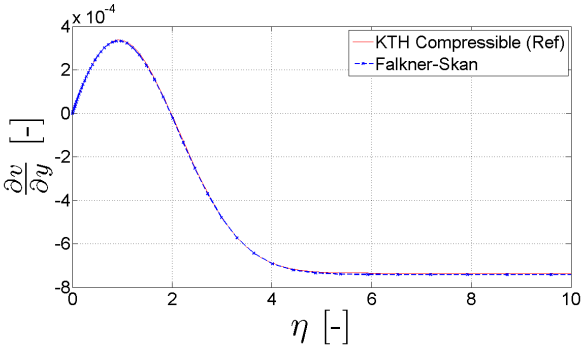
(a) Streamwise velocity u



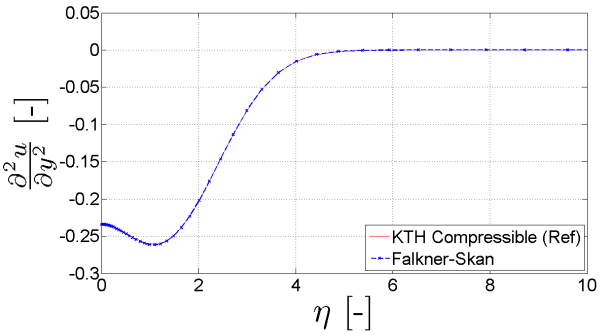
(b) Wall-normal velocity v



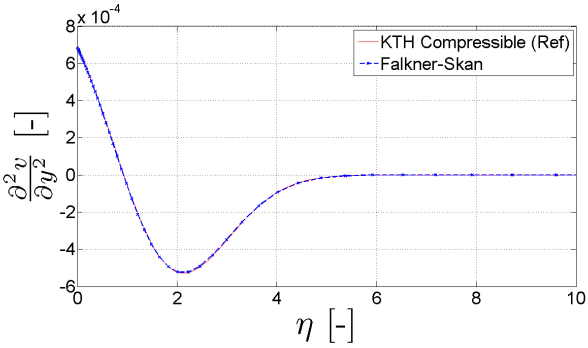
(c) Wall-normal derivative of u



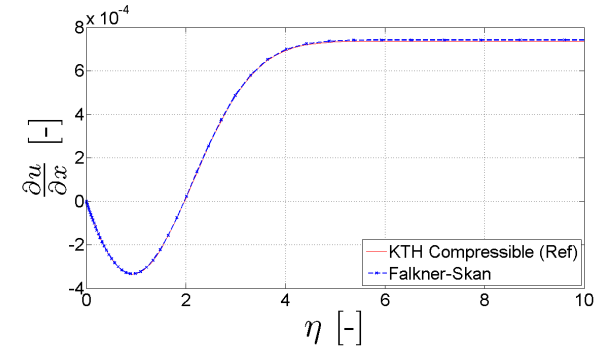
(d) Wall-normal derivative of v



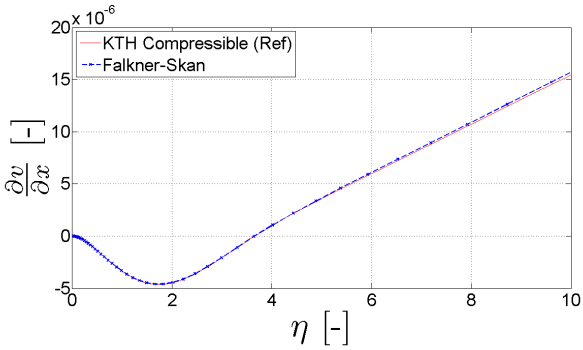
(e) Second wall-normal derivative of u



(f) Second wall-normal derivative of v

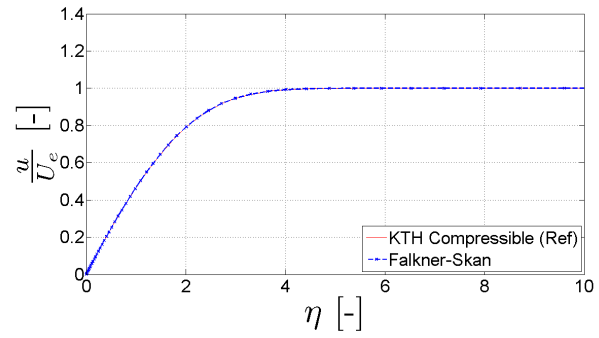


(g) Streamwise derivative of u

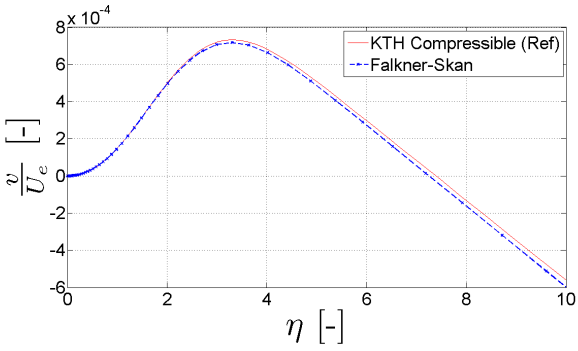


(h) Streamwise derivative of v

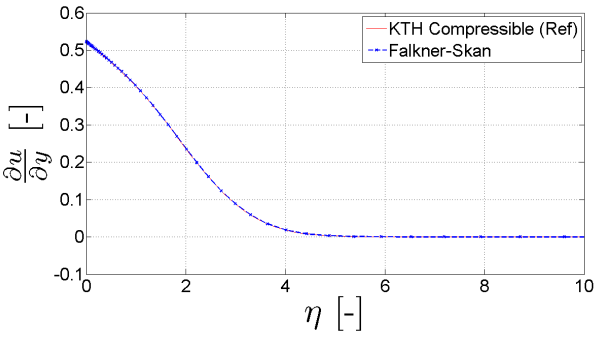
Figure 3.12.: Boundary-layer profiles of the pressure side at $x_c = 0.0594$ for the Falkner-Skan transformation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).



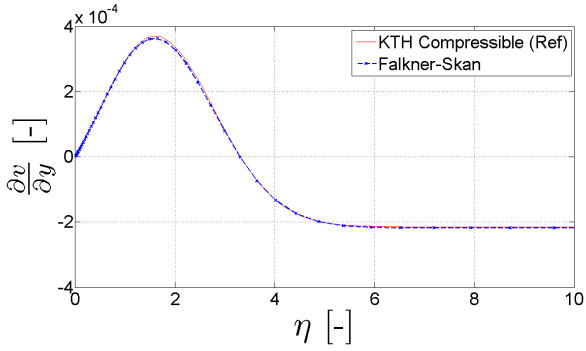
(a) Streamwise velocity u



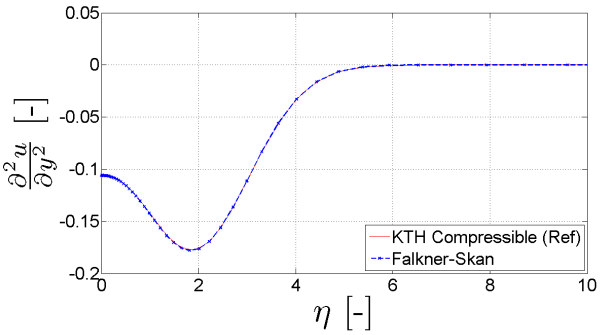
(b) Wall-normal velocity v



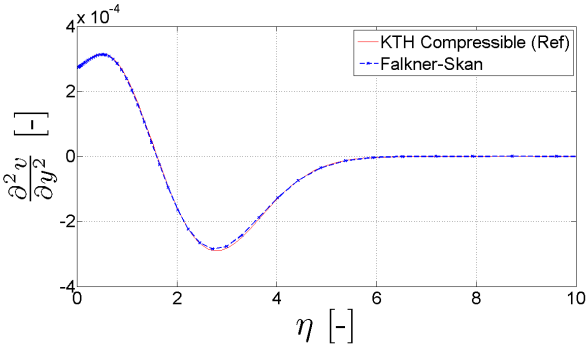
(c) Wall-normal derivative of u



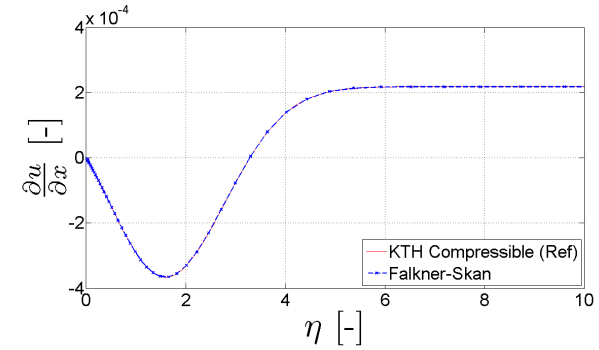
(d) Wall-normal derivative of v



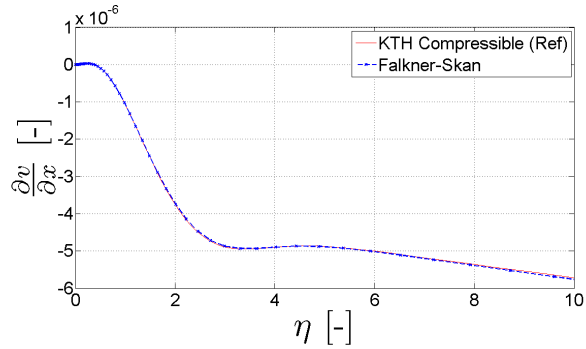
(e) Second wall-normal derivative of u



(f) Second wall-normal derivative of v



(g) Streamwise derivative of u

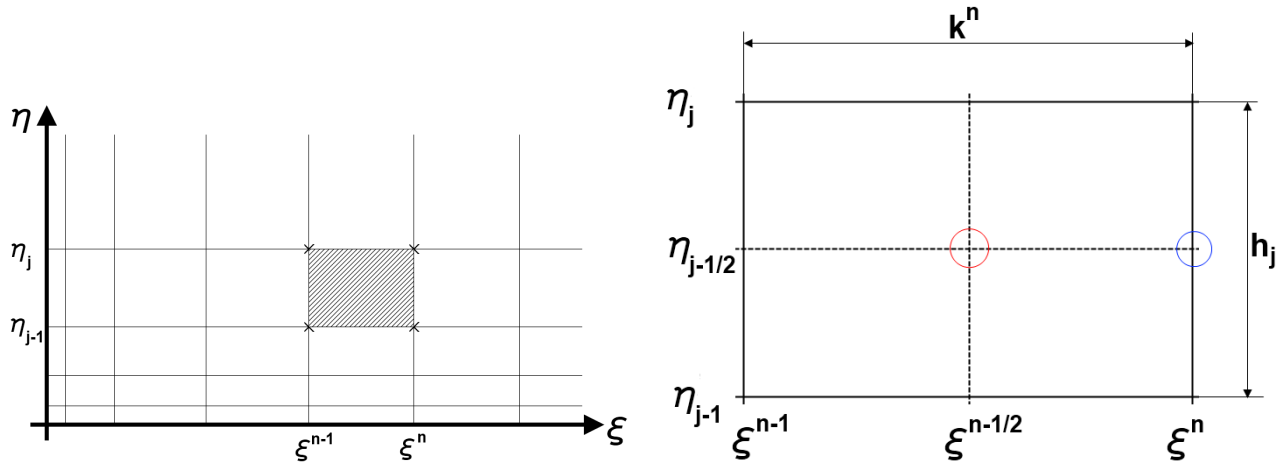


(h) Streamwise derivative of v

Figure 3.13.: Boundary-layer profiles of the suction side at $x_c = 0.0373$ for the Falkner-Skan transformation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

3.4.2 Discretization of the Boundary-Layer Equation using the Keller Box Method

In contrast to subsection 3.4.1 a different type of discretization is used in the following reducing the dependence on the grid by employing four instead of two grid points to set up the differentials. A sketch of the Keller box is shown in Figure 3.14 where equations containing derivatives in streamwise and wall-normal direction are evaluated at the center of the Keller box (red circle in Figure 3.14b) and equations containing only derivatives in wall-normal direction are evaluated at the nth position (blue circle in Figure 3.14b).



(a) Rectangle for Keller Box **(b)** Center Scheme

Figure 3.14.: Sketches for the derivation of the differencing schemes using the Keller box.

This discretization is first applied to the Falkner-Skan transformation and then to the Görtler transformation.

The Falkner-Skan Transformation

As already described in section 3.3 the Falkner-Skan equation (2.75), a third-order differential equation in wall-normal direction, is reduced to a system of three differential equations of first order

$$\frac{\partial f}{\partial \eta} = f' = u \tag{3.50a}$$

$$\frac{\partial^2 f}{\partial \eta^2} = u' = v \tag{3.50b}$$

$$\frac{\partial^3 f}{\partial \eta^3} = v' = -\frac{m+1}{2} f v - m\left[1-u^2\right] + x\left[u\frac{\partial u}{\partial x} - v\frac{\partial f}{\partial x}\right]. \tag{3.50c}$$

Applying a centering around the nth location for 3.50a and 3.50b and a centering around the midpoint for 3.50c it follows

$$\frac{f_{(j)}^{(n)} - f_{(j-1)}^{(n)}}{h_{(j)}} = \frac{u_{(j)}^{(n)} + u_{(j-1)}^{(n)}}{2} \equiv u_{(j-\frac{1}{2})}^{(n)} \tag{3.51a}$$

$$\frac{u_{(j)}^{(n)} - u_{(j-1)}^{(n)}}{h_{(j)}} = \frac{v_{(j)}^{(n)} + v_{(j-1)}^{(n)}}{2} \equiv v_{(j-\frac{1}{2})}^{(n)} \tag{3.51b}$$

$$\frac{v_{(j)}^{(n-\frac{1}{2})} - v_{(j-1)}^{(n-\frac{1}{2})}}{h_{(j)}} + \frac{m^{(n-\frac{1}{2})}+1}{2}\left(fv\right)_{(j-\frac{1}{2})}^{(n-\frac{1}{2})} + \\ + m^{(n-\frac{1}{2})}\left[1 - \left(u^2\right)_{(j-\frac{1}{2})}^{(n-\frac{1}{2})}\right] = x^{(n-\frac{1}{2})}\left[u_{(j-\frac{1}{2})}^{(n-\frac{1}{2})}\frac{u_{(j-\frac{1}{2})}^{(n)} - u_{(j-\frac{1}{2})}^{(n-1)}}{k^{(n)}} - v_{(j-\frac{1}{2})}^{(n-\frac{1}{2})}\frac{f_{(j-\frac{1}{2})}^{(n)} - f_{(j-\frac{1}{2})}^{(n-1)}}{k^{(n)}}\right].$$

In order to solve the system some rearrangements have to be done. The terms in the last equation are sorted such that current positions (terms including n) will be on the left-hand side, while terms that contain only previous positions or other terms that are known are moved to the right. The following substitutions are applied

$$\phi_{(j)}^{(n-\frac{1}{2})} = \frac{\phi_{(j)}^{(n)} - \phi_{(j)}^{(n-1)}}{k^{(n)}} = \frac{\phi_{(j)}^{(n)} + \phi_{(j)}^{(n-1)}}{2}$$
$$\phi_{(j-\frac{1}{2})}^{(n)} = \frac{\phi_{(j)}^{(n)} - \phi_{(j-1)}^{(n)}}{h_{(j)}} = \frac{\phi_{(j)}^{(n)} + \phi_{(j-1)}^{(n)}}{2}$$

where ϕ is the variable for which the substitution is used. By applying those substitutions, it follows

$$\frac{v_{(j)}^{(n)} - v_{(j-1)}^{(n)}}{h_{(j)}} + \frac{m^{(n-\frac{1}{2})}+1}{2}\left(fv\right)_{(j-\frac{1}{2})}^{(n)} - m^{(n-\frac{1}{2})}\left(u^2\right)_{(j-\frac{1}{2})}^{(n)} - \\ - \frac{x^{(n-\frac{1}{2})}}{k^{(n)}}\left[\left(u^2\right)_{(j-\frac{1}{2})}^{(n)} - \left(fv\right)_{(j-\frac{1}{2})}^{(n)} - f_{(j-\frac{1}{2})}^{(n)}v_{(j-\frac{1}{2})}^{(n-1)} + f_{(j-\frac{1}{2})}^{(n-1)}v_{(j-\frac{1}{2})}^{(n)}\right] = R_{(j-\frac{1}{2})}^{(n-1)} \tag{3.51c}$$

where the known terms and terms of the previous position are summarized in

$$R_{(j-\frac{1}{2})}^{(n-1)} = -L_{(j-\frac{1}{2})}^{(n-1)} + \frac{x^{(n-\frac{1}{2})}}{k^{(n)}}\left[-\left(u^2\right)_{(j-\frac{1}{2})}^{(n-1)} + \left(fv\right)_{(j-\frac{1}{2})}^{(n-1)}\right] \tag{3.52a}$$
$$L_{(j-\frac{1}{2})}^{(n-1)} = \frac{v_{(j)}^{(n-1)} - v_{(j-1)}^{(n-1)}}{h_{(j)}} + \frac{m^{(n-\frac{1}{2})}+1}{2}\left(fv\right)_{(j-\frac{1}{2})}^{(n-1)} + m^{(n-\frac{1}{2})}\left[2 - \left(u^2\right)_{(j-\frac{1}{2})}^{(n-1)}\right]. \tag{3.52b}$$

In a next step a Newton method is introduced to solve equations 3.51. This is necessary as the discretization is still a system of nonlinear equations that have to be solved iteratively. The terms f, u and v are approximated as

$$f_{(j)}^{(n),(i+1)} = f_{(j)}^{(n),(i)} + \delta f_{(j)}^{(n),(i)}, \tag{3.53a}$$
$$u_{(j)}^{(n),(i+1)} = u_{(j)}^{(n),(i)} + \delta u_{(j)}^{(n),(i)}, \tag{3.53b}$$
$$v_{(j)}^{(n),(i+1)} = v_{(j)}^{(n),(i)} + \delta v_{(j)}^{(n),(i)} \tag{3.53c}$$

where i is the iteration number. As the boundary-layer equations are nonlinear the system has to be linearized which is done by neglecting all quadratic terms of $\delta f_{(j)}^{(n),(i)}$, $\delta u_{(j)}^{(n),(i)}$ and $\delta v_{(j)}^{(n),(i)}$. In the following

the superscript n for an unknown variable will be dropped due to reason of simplicity. Then, the system of equations (3.51) expands to

$$
\begin{aligned}
f_{(j)}^{(i)} + \delta f_{(j)}^{(i)} - f_{(j-1)}^{(i)} - \delta f_{(j-1)}^{(i)} &= \frac{h_{(j)}}{2}\left(u_{(j)}^{(i)} + \delta u_{(j)}^{(i)} + u_{(j-1)}^{(i)} + \delta u_{(j-1)}^{(i)}\right) \\
u_{(j)}^{(i)} + \delta u_{(j)}^{(i)} - u_{(j-1)}^{(i)} - \delta u_{(j-1)}^{(i)} &= \frac{h_{(j)}}{2}\left(v_{(j)}^{(i)} + \delta v_{(j)}^{(i)} + v_{(j-1)}^{(i)} + \delta v_{(j-1)}^{(i)}\right) \\
\frac{v_{(j)}^{(i)} - v_{(j-1)}^{(i)}}{h_{(j)}} + \frac{m^{(n-\frac{1}{2})}+1}{2}\left(f\,v\right)_{(j-\frac{1}{2})}^{(i)} - m^{(n-\frac{1}{2})}\left(u^2\right)_{(j-\frac{1}{2})}^{(i)} - \\
- \frac{x^{(n-\frac{1}{2})}}{k^{(n)}}\left[\left(u^2\right)_{(j-\frac{1}{2})}^{(n)} - \left(f\,v\right)_{(j-\frac{1}{2})}^{(i)} - f_{(j-\frac{1}{2})}^{(i)} v_{(j-\frac{1}{2})}^{(n-1)} + f_{(j-\frac{1}{2})}^{(n-1)} v_{(j-\frac{1}{2})}^{(i)}\right] + \frac{\delta v_{(j)}^{(i)} - \delta v_{(j-1)}^{(i)}}{h_{(j)}} + \\
+ \frac{1}{2}\frac{m^{(n-\frac{1}{2})}+1}{2}\left(f_{(j-\frac{1}{2})}^{(i)}\delta v_{(j)}^{(i)} + f_{(j-\frac{1}{2})}^{(i)}\delta v_{(j-1)}^{(i)} + v_{(j-\frac{1}{2})}^{(i)}\delta f_{(j)}^{(i)} + v_{(j-\frac{1}{2})}^{(i)}\delta f_{(j-1)}^{(i)}\right) - \\
- m^{(n-\frac{1}{2})}\left(u_{(j-\frac{1}{2})}^{(i)}\delta u_{(j)}^{(i)} + u_{(j-\frac{1}{2})}^{(i)}\delta u_{(j-1)}^{(i)}\right) - \frac{x^{(n-\frac{1}{2})}}{k^{(n)}}\left[u_{(j-\frac{1}{2})}^{(i)}\delta u_{(j)}^{(i)} + u_{(j-\frac{1}{2})}^{(i)}\delta u_{(j-1)}^{(i)} - \right. \\
- \frac{1}{2}\left(\overset{(i)}{}_{(j-\frac{1}{2})}\delta v_{(j)}^{(i)} + f_{(j-\frac{1}{2})}^{(i)}\delta v_{(j)}^{(i-1)} + v_{(j-\frac{1}{2})}^{(i)}\delta f_{(j)}^{(i)} + v_{(j-\frac{1}{2})}^{(i)}\delta f_{(j)}^{(i-1)} + \right. \\
\left.\left. + v_{(j-\frac{1}{2})}^{(n-1)}\delta f_{(j)}^{(i)} + v_{(j-\frac{1}{2})}^{(n-1)}\delta f_{(j-1)}^{(i)} - f_{(j-\frac{1}{2})}^{(n-1)}\delta v_{(j)}^{(i)} - f_{(j-\frac{1}{2})}^{(n-1)}\delta v_{(j-1)}^{(i)}\right)\right] = R_{(j-\frac{1}{2})}^{(n-1)}
\end{aligned}
$$

which can be summarized as

$$\delta f_{(j)}^{(i)} - \delta f_{(j-1)}^{(i)} - \frac{h_{(j)}}{2}\left(\delta u_{(j)}^{(i)} + \delta u_{(j-1)}^{(i)}\right) = \left(r_1\right)_{(j)} \tag{3.54a}$$

$$\delta u_{(j)}^{(i)} - \delta u_{(j-1)}^{(i)} - \frac{h_{(j)}}{2}\left(\delta v_{(j)}^{(i)} + \delta v_{(j-1)}^{(i)}\right) = \left(r_3\right)_{(j-1)} \tag{3.54b}$$

$$
\begin{aligned}
\left(s_1\right)_{(j)}\delta v_{(j)}^{(i)} + \left(s_2\right)_{(j)}\delta v_{(j-1)}^{(i)} + \left(s_3\right)_{(j)}\delta f_{(j)}^{(i)} + \\
+ \left(s_4\right)_{(j)}\delta f_{(j-1)}^{(i)} + \left(s_5\right)_{(j)}\delta u_{(j)}^{(i)} + \left(s_6\right)_{(j)}\delta u_{(j-1)}^{(i)} = \left(r_2\right)_{(j)}
\end{aligned} \tag{3.54c}
$$

with

$$\left(r_1\right)_{(j)} = -\left(f_{(j)}^{(i)} - f_{(j-1)}^{(i)}\right) + h_{(j)}u_{(j-\frac{1}{2})}^{(i)} \tag{3.55a}$$

$$\left(r_3\right)_{(j-1)} = -\left(u_{(j)}^{(i)} - u_{(j-1)}^{(i)}\right) + h_{(j)}v_{(j-\frac{1}{2})}^{(i)} \tag{3.55b}$$

$$
\begin{aligned}
\left(r_2\right)_{(j)} = R_{(j-\frac{1}{2})}^{(n-1)} - \left[\frac{v_{(j)}^{(i)} - v_{(j-1)}^{(i)}}{h_{(j)}} + \alpha_1^{(n)}\left(f\,v\right)_{(j-\frac{1}{2})}^{(i)} - \alpha_2^{(n)}\left(u^2\right)_{(j-\frac{1}{2})}^{(n)} + \right. \\
\left. + \alpha^{(n)}\left(f_{(j-\frac{1}{2})}^{(i)}v_{(j-\frac{1}{2})}^{(n-1)} - f_{(j-\frac{1}{2})}^{(n-1)}v_{(j-\frac{1}{2})}^{(i)}\right)\right]
\end{aligned} \tag{3.55c}
$$

and

$$(s_1)_{(j)} = \frac{1}{h_{(j)}} + \frac{\alpha_1^{(n)}}{2} f^{(i)}_{(j-\frac{1}{2})} - \frac{\alpha^{(n)}}{2} f^{(n-1)}_{(j-\frac{1}{2})} \tag{3.56a}$$

$$(s_2)_{(j)} = -\frac{1}{h_{(j)}} + \frac{\alpha_1^{(n)}}{2} f^{(i)}_{(j-\frac{1}{2})} - \frac{\alpha^{(n)}}{2} f^{(n-1)}_{(j-\frac{1}{2})} \tag{3.56b}$$

$$(s_3)_{(j)} = \frac{\alpha_1^{(n)}}{2} v^{(i)}_{(j-\frac{1}{2})} + \frac{\alpha^{(n)}}{2} v^{(n-1)}_{(j-\frac{1}{2})} \tag{3.56c}$$

$$(s_4)_{(j)} = \frac{\alpha_1^{(n)}}{2} v^{(i)}_{(j-\frac{1}{2})} + \frac{\alpha^{(n)}}{2} v^{(n-1)}_{(j-\frac{1}{2})} \tag{3.56d}$$

$$(s_5)_{(j)} = -\alpha_2^{(n)} u^{(i)}_{(j-\frac{1}{2})} \tag{3.56e}$$

$$(s_6)_{(j)} = -\alpha_2^{(n)} u^{(i)}_{(j-\frac{1}{2})}. \tag{3.56f}$$

The new variables which are introduced here are the constants

$$\alpha^{(n)} = \frac{x^{(n-\frac{1}{2})}}{k^{(n)}} \tag{3.57a}$$

$$\alpha_1^{(n)} = \frac{m^{(n-\frac{1}{2})} + 1}{2} + \alpha^{(n)} \tag{3.57b}$$

$$\alpha_2^{(n)} = m^{(n-\frac{1}{2})} + \alpha^{(n)} \tag{3.57c}$$

that do not depend on the wall-normal direction η. Note that all values at the previous position $n-1$ are known for $0 \leq j \leq J$ and only values which have the superscript i are solved for. That means the system has $3J+3$ unknowns for $\delta f^{(n),(i)}_{(j)}$, $\delta u^{(n),(i)}_{(j)}$ and $\delta v^{(n),(i)}_{(j)}$ with $j = 0, 1, \ldots J$. The values for $(s_1)_{(j)}$ to $(s_6)_{(j)}$ and $(r_1)_{(j)}$, $(r_2)_{(j)}$ and $(r_3)_{(j-1)}$ are calculated for $j = 1, 2, \ldots J$ and thus three additional conditions have to be specified which are the boundary conditions

$$\delta f^{(i)}_{(0)} = 0 \tag{3.58a}$$

$$\delta u^{(i)}_{(0)} = 0 \tag{3.58b}$$

$$\delta u^{(i)}_{(J)} = 0. \tag{3.58c}$$

Now, the linearized system of equations can be solved. A very efficient way to solve the system is to create a tridiagonal block structured matrix and solve it with the Thomas algorithm. The system of equation in matrix form reads

$$\boldsymbol{A}^{(i)} \boldsymbol{\delta}^{(i)} = \boldsymbol{r}^{(i)} \tag{3.59}$$

where $\boldsymbol{A}^{(i)}$ has block-tridiagonal form. In detail (3.59) becomes

$$\begin{pmatrix} \boldsymbol{A}_{(0)}^{(i)} & \boldsymbol{C}_{(0)}^{(i)} & & & & & & \\ \boldsymbol{B}_{(1)}^{(i)} & \boldsymbol{A}_{(1)}^{(i)} & \boldsymbol{C}_{(1)}^{(i)} & & & & & \\ & \ddots & \ddots & \ddots & & & & \\ & & \boldsymbol{B}_{(j)}^{(i)} & \boldsymbol{A}_{(j)}^{(i)} & \boldsymbol{C}_{(j)}^{(i)} & & & \\ & & & \ddots & \ddots & \ddots & & \\ & & & & \boldsymbol{B}_{(J-1)}^{(i)} & \boldsymbol{A}_{(J-1)}^{(i)} & \boldsymbol{C}_{(J-1)}^{(i)} \\ & & & & & \boldsymbol{B}_{(J)}^{(i)} & \boldsymbol{A}_{(J)}^{(i)} \end{pmatrix} \begin{pmatrix} \vec{\delta}_{(0)}^{(i)} \\ \vec{\delta}_{(1)}^{(i)} \\ \vdots \\ \vec{\delta}_{(j)}^{(i)} \\ \vdots \\ \vec{\delta}_{(J-1)}^{(i)} \\ \vec{\delta}_{(J)}^{(i)} \end{pmatrix} = \begin{pmatrix} \vec{r}_{(0)}^{(i)} \\ \vec{r}_{(1)}^{(i)} \\ \vdots \\ \vec{r}_{(j)}^{(i)} \\ \vdots \\ \vec{r}_{(J-1)}^{(i)} \\ \vec{r}_{(J)}^{(i)} \end{pmatrix} \tag{3.60}$$

where $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are 3x3 matrices defined as

$$\boldsymbol{A}_{(0)}^{(i)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & -\frac{h_{(1)}}{2} \end{pmatrix} \qquad \boldsymbol{A}_{(j)}^{(i)} = \begin{pmatrix} 1 & -\frac{h_{(j)}}{2} & 0 \\ (s_3)_{(j)} & (s_5)_{(j)} & (s_1)_{(j)} \\ 0 & -1 & -\frac{h_{(j+1)}}{2} \end{pmatrix} \quad \text{for} \quad 1 \leq j \leq J-1$$

$$\boldsymbol{A}_{(J)}^{(i)} = \begin{pmatrix} 1 & -\frac{h_{(J)}}{2} & 0 \\ (s_3)_{(J)} & (s_5)_{(J)} & (s_1)_{(J)} \\ 0 & 1 & 0 \end{pmatrix} \tag{3.61a}$$

$$\boldsymbol{B}_{(j)}^{(i)} = \begin{pmatrix} -1 & -\frac{h_{(j)}}{2} & 0 \\ (s_4)_{(j)} & (s_6)_{(j)} & (s_2)_{(j)} \\ 0 & 0 & 0 \end{pmatrix} \quad \text{for} \quad 1 \leq j \leq J \tag{3.61b}$$

$$\boldsymbol{C}_{(j)}^{(i)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & -\frac{h_{(j+1)}}{2} \end{pmatrix} \quad \text{for} \quad 0 \leq j \leq J-1 \tag{3.61c}$$

and

$$\vec{\delta}_{(j)}^{(i)} = \begin{pmatrix} \delta f_{(j)}^{(i)} \\ \delta u_{(j)}^{(i)} \\ \delta v_{(j)}^{(i)} \end{pmatrix} \quad \text{for} \quad 0 \leq j \leq J \tag{3.62a}$$

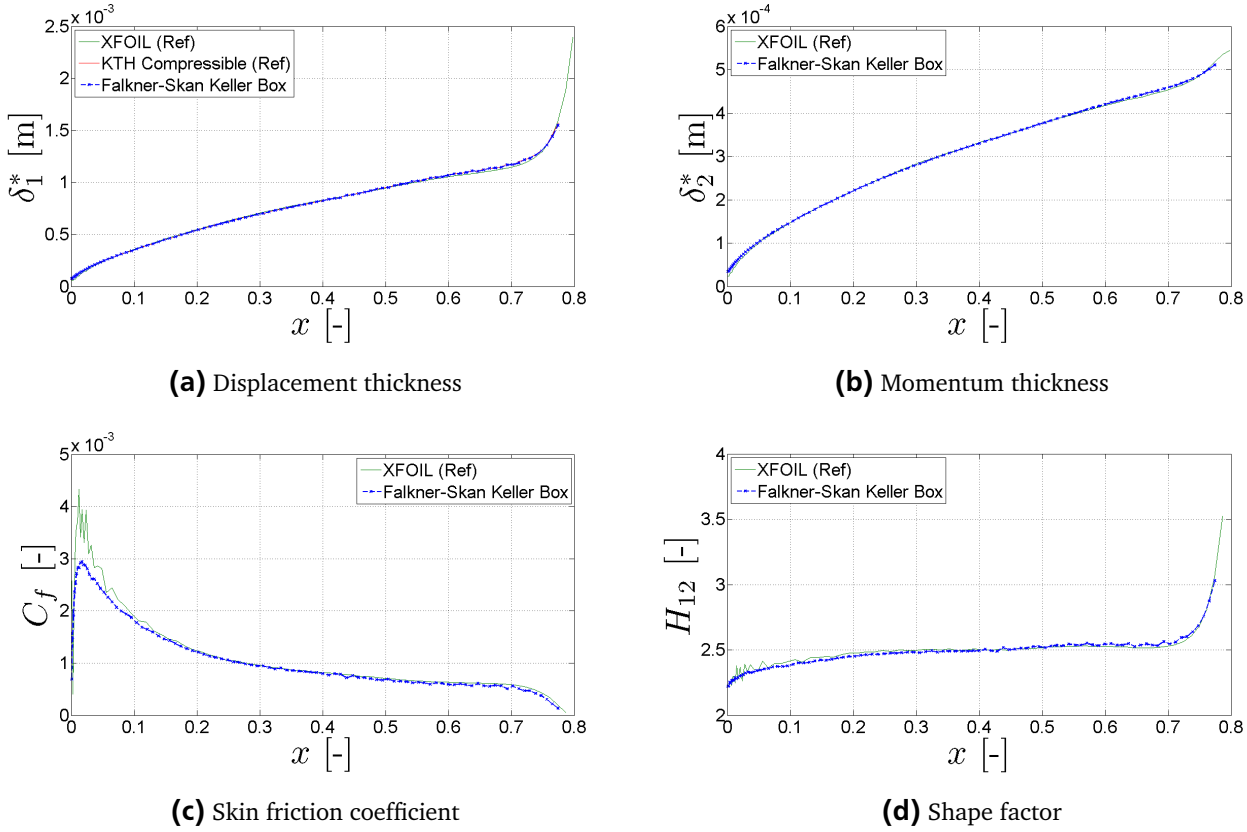
$$\vec{r}_{(j)}^{(i)} = \begin{pmatrix} (r_1)_{(j)} \\ (r_2)_{(j)} \\ (r_3)_{(j)} \end{pmatrix} \quad \text{for} \quad 0 \leq j \leq J. \tag{3.62b}$$

Note that the first two rows and the last row of (3.60) correspond to the boundary conditions (3.58). Therefore, it follows that $(r_1)_{(0)} = 0$, $(r_2)_{(0)} = 0$ and $(r_3)_{(J)} = 0$ in (3.62b) to ensure that the boundary conditions (3.58) are fulfilled. Arranging the system of equation in the order given above ensures that the matrix $\boldsymbol{A}_{(0)}$ does not become singular [10]. Due to the tridiagonal block form the system can easily and quickly be solved with *MATLAB* by defining $\boldsymbol{A}$ to be a sparse matrix. In *MATLAB* this is done with the command `spalloc`. The algorithm that is applied for those special cases is the Thomas algorithm that is described in section 3.1.4.

After the solution is converged the velocity profiles and their derivatives are calculated by using the

laminar equations given in 2.74. Those profiles can be used for transition prediction by stability analysis afterwards. Further, the boundary-layer characteristics are computed which can be used for a viscous-inviscid interaction method. Finally, the calculation is completed for the current streamwise position and the procedure is repeated for all streamwise positions until the separation point where no convergence is achieved.

In the following the results for a calculation of the *MW-airfoil* are shown. The angle of attack for the calculation is $\alpha = 3.3°$ for a Reynolds number of $Re = 2\,351\,842$ and a Mach number of $Ma = 0.01$. The Mach number is used to compute the free-stream velocity U^*_∞. The converged solutions for the pressure and the suction side are depicted in Figures 3.15 to 3.18. Note that all profile plots are scaled with the Blasius length scale as length scale and the edge velocity at the boundary layer U^*_e as velocity scale.



(a) Displacement thickness

(b) Momentum thickness

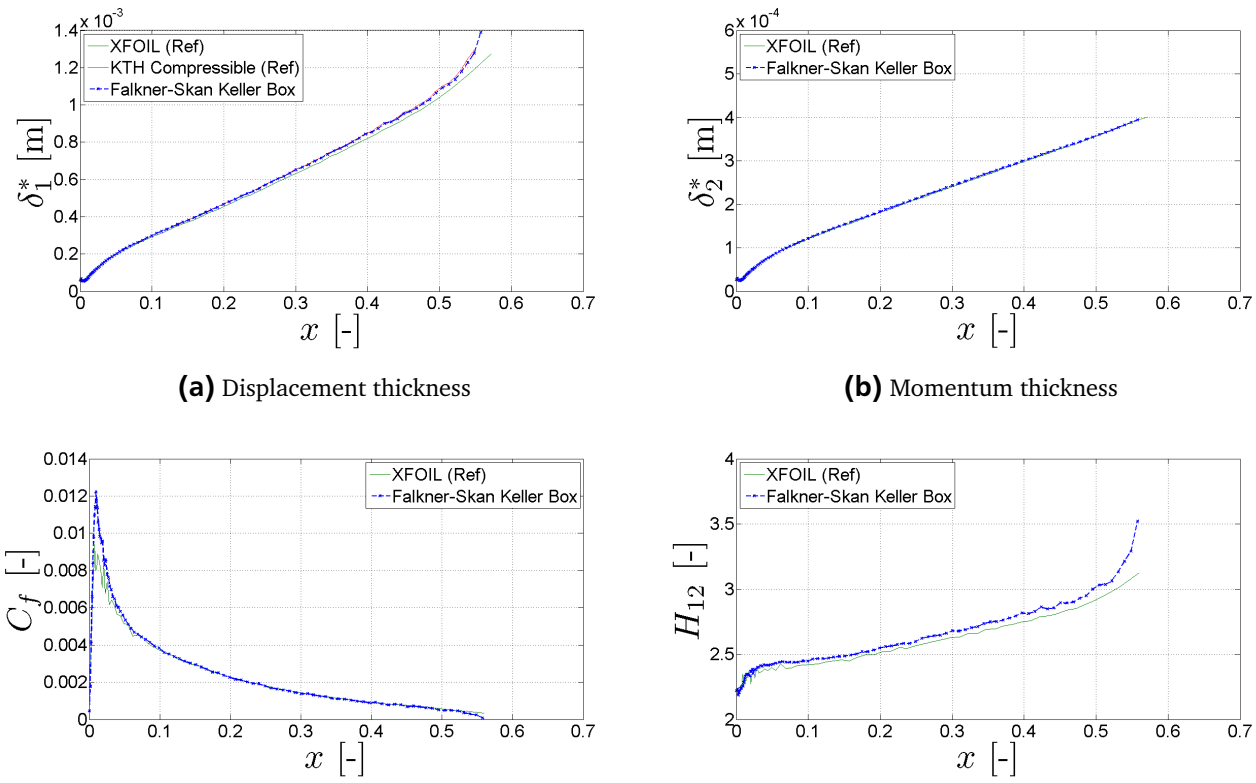
(c) Skin friction coefficient

(d) Shape factor

Figure 3.15.: Boundary-layer characteristics of the pressure side for the Falkner-Skan transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

The boundary-layer characteristics for the pressure side as shown in Figure 3.15 match quite well with the results from *XFOIL*. The trend is detected in each plot whereas the absolute values do not match perfectly. The reason for the discrepancy is that the solution of *XFOIL* is based on a converged solution of a viscous-inviscid interactive method. In order words, the pressure distribution found by *XFOIL* for the converged solution is slightly different resulting in slightly different integral values. However, the solution matches perfectly with the reference data of the KTH solver is shown in Figure 3.15a that is based on the same pressure distribution. Further, the numerical scheme using the Keller box seems to be very stable as the curves are very smooth. Especially, in the case of the friction coefficient (Figure 3.15c) the vibrations vanish completely. Further, the onset of transition is detected very well which is characterized by an increase of the shape factor. The value of the separation point where the skin friction

becomes $C_f = 0$ will be detected as well. The last converged solution is found for $x = 0.7743$ which is close to that separation point.
The characteristics for the suction side are illustrated in Figure 3.16 and again the overall trend matches with the results from *XFOIL*. Obviously, the change of the converged pressure distribution is larger for the suction side as the discrepancy between the absolute values is larger than for the pressure side. However, the results match perfectly with the data of the KTH solver as shown in Figure 3.16a. As clearly indicated in Figure 3.16d the onset of transition is computed too early because the shape factor rises earlier. However, that can be explained with the different pressure distributions. Likewise for the pressure side that numerical behaves very stable is vibrations are damped as depicted in Figure 3.16c. In that figure the vanishing skin friction is also found earlier as already seen in the case of the shape factor. In contrast to the pressure distribution the skin friction coefficient is overestimated due to the different pressure distribution. The decrease of displacement thickness and momentum thickness in Figures 3.16a and 3.16b at the beginning marks the positions from the stagnation point to the leading edge. In those regions the profiles become “fuller” resulting in decreasing values. After the leading edge the flow accelerates and the values increase again.



(a) Displacement thickness

(b) Momentum thickness

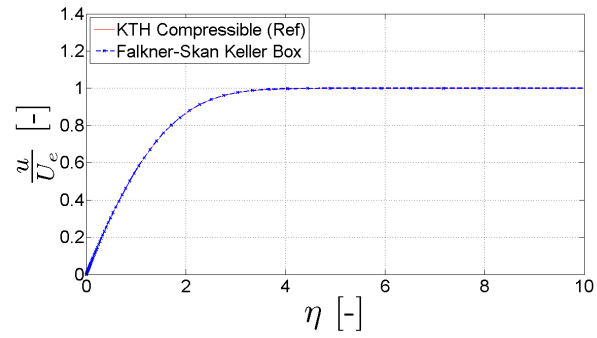
(c) Skin friction coefficient

(d) Shape factor

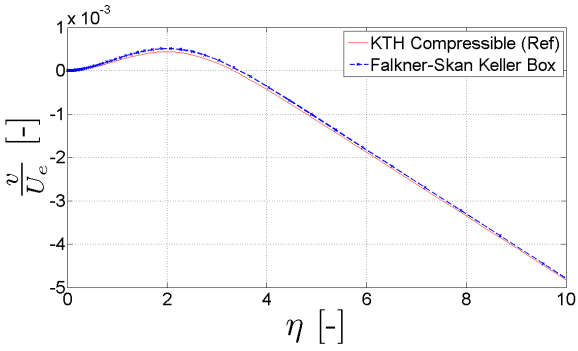
Figure 3.16.: Boundary-layer characteristics of the suction side for the Falkner-Skan transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

The boundary-layer profiles for the pressure side as shown in Figure 3.17 are computed for $x = 0.0584$ which corresponds to $x_c = 0.0594$. In general the qualitative behavior is detected and the agreement is better for the streamwise velocity component than for the wall-normal component. The u velocity and its derivatives in wall-normal direction match perfectly with the reference data as shown in Figures 3.17a, 3.17c and 3.17e. Those profiles are three to five orders of magnitude larger than the other profiles which could be one reason for the good agreement. Further, inputs needed to calculate those profiles

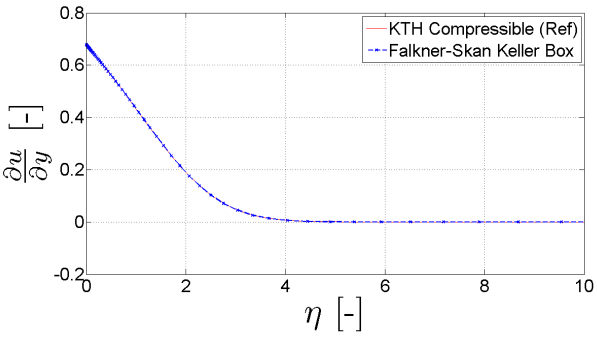
are directly obtained from the numerical scheme. In contrast to that the other profiles are slightly overestimated. The inputs were calculated using differencing schemes as described in section 3.1.3 and thus slight differences could be possible. Whereas the differences are small for the v component as shown in Figure 3.17b they increase in the derivatives as more terms are needed as input that do not directly come from the numerical scheme (Figures 3.17d and 3.17f). The only profile that does not match with the reference is the streamwise derivative of v as depicted in Figure 3.17h. A possible explanation could be the discretization of the Keller box as the values are centered at the middle of two grid points. As the position of the profiles is quite close to the leading edge large differences for the pressure distribution of the grid points can be assumed which might lead to a different result.
Similar results are found for the boundary-layer profiles of the suction side as shown in Figure 3.18. The position where the profiles are taken is $x = 0.0689$ which corresponds to $x_c = 0.0373$. Like for the profiles of the pressure side the agreement is better for profiles containing the streamwise velocity u. Whereas the profiles for u and its wall-normal derivative match perfectly with the reference data (Figure 3.18a and 3.18c) the second derivative has a small bump near the wall as shown in Figure 3.18e. The difference could be explained with the different discretization as the profiles are again taken very close to the leading edge. The discrepancies for the wall-normal velocity component are larger than for the pressure side which can also be traced back to the different discretization technique. As the differences in the pressure distribution are quite large near the leading edge the differences increase in the profiles. Especially the profiles with low order of magnitude differ noticeably as depicted in Figures 3.18b, 3.18d and 3.18f. In Figure 3.18f the first peak near the wall is not detected at all. In contrast to the pressure side the profile for the streamwise derivative of v as shown in Figure 3.18h at least the qualitative behavior is detected but the absolute values are different.
The qualitative behavior is similar for other positions along the airfoil for the suction and pressure side. For positions with smaller pressure gradients the profiles are closer to the KTH reference.



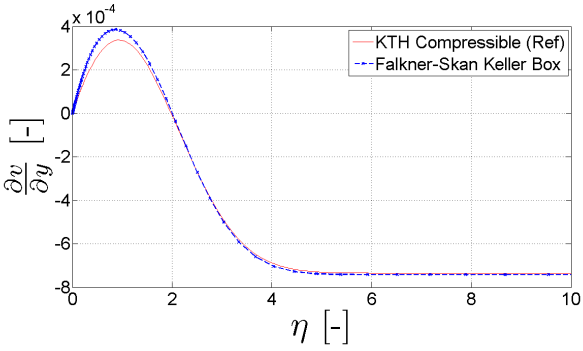
(a) Streamwise velocity u



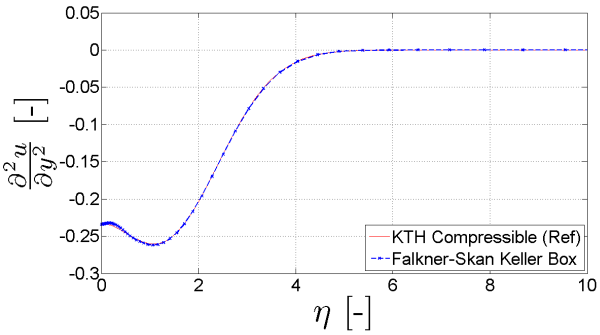
(b) Wall-normal velocity v



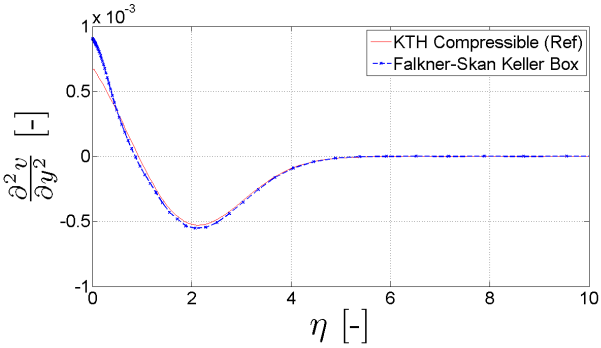
(c) Wall-normal derivative of u



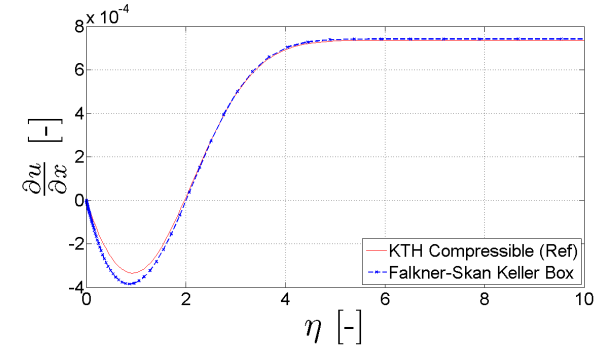
(d) Wall-normal derivative of v



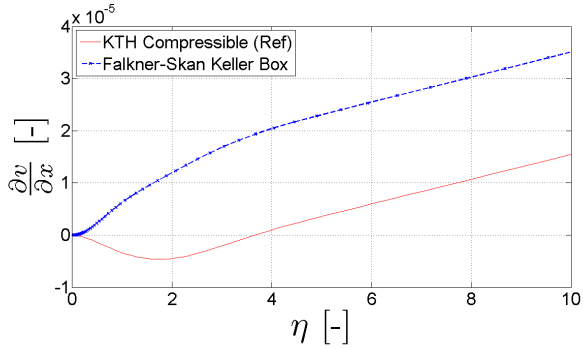
(e) Second wall-normal derivative of u



(f) Second wall-normal derivative of v

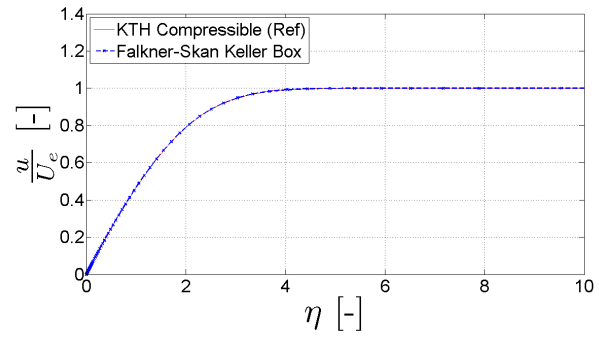


(g) Streamwise derivative of u

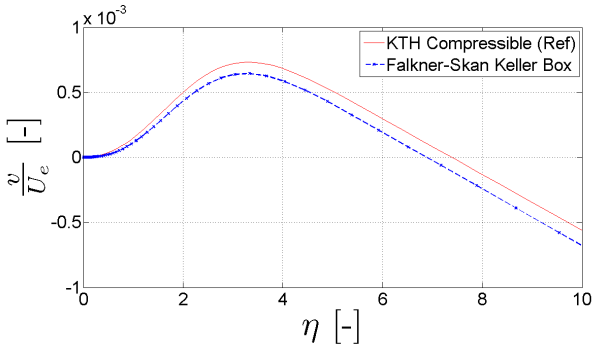


(h) Streamwise derivative of v

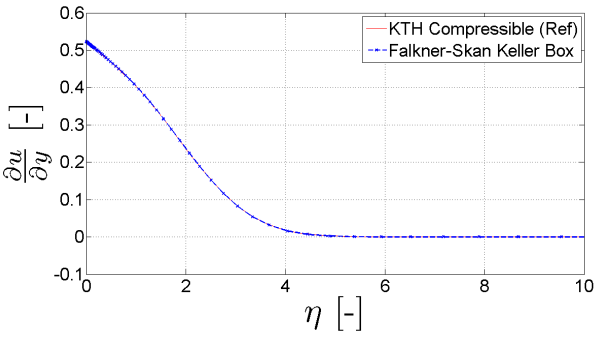
Figure 3.17.: Boundary-layer profiles of the pressure side at $x_c = 0.0594$ for the Falkner-Skan transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).



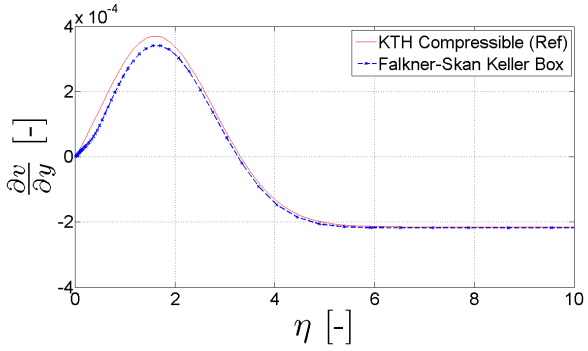
(a) Streamwise velocity u



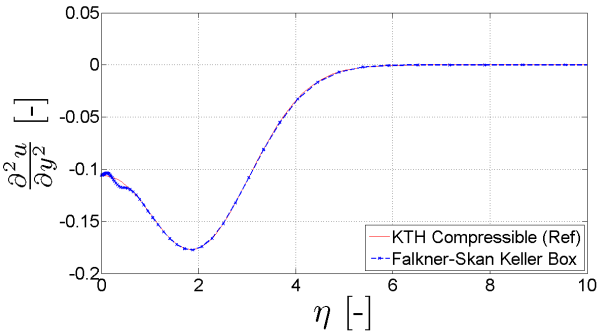
(b) Wall-normal velocity v



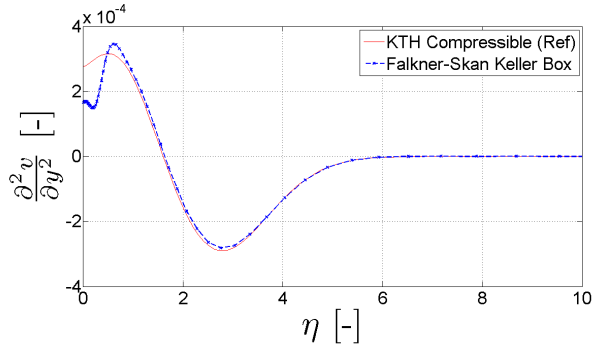
(c) Wall-normal derivative of u



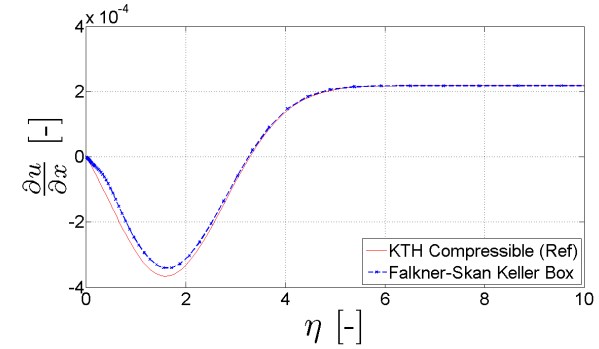
(d) Wall-normal derivative of v



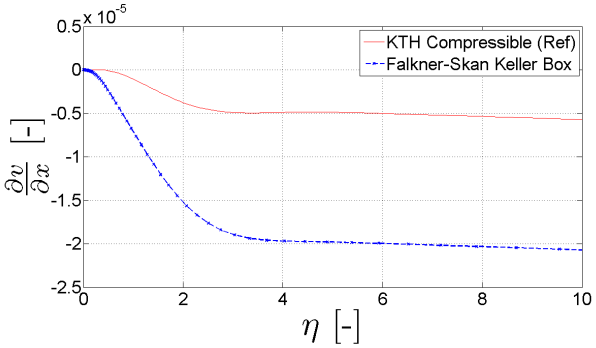
(e) Second wall-normal derivative of u



(f) Second wall-normal derivative of v



(g) Streamwise derivative of u



(h) Streamwise derivative of v

Figure 3.18.: Boundary-layer profiles of the suction side at $x_c = 0.0373$ for the Falkner-Skan transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

The the same way as described in above's subsection for the Falkner-Skan transformation the Keller box is applied to the Görtler transformation (2.97). Therefore, the transformed boundary-layer equation is expressed in a system of three first order differential equations

$$\frac{\partial F}{\partial \eta} = F' = U \tag{3.63a}$$

$$\frac{\partial^2 F}{\partial \eta^2} = U' = V \tag{3.63b}$$

$$\frac{\partial^3 F}{\partial \eta^3} = V' = -FV - \beta(\xi)\left[1 - U^2\right] + 2\xi\left[U\frac{\partial U}{\partial \xi} - V\frac{\partial F}{\partial \xi}\right]. \tag{3.63c}$$

Using the centering of the Keller box the system (3.63) is discretized such that

$$\frac{F_{(j)}^{(n)} - F_{(j-1)}^{(n)}}{h_{(j)}} = \frac{U_{(j)}^{(n)} + U_{(j-1)}^{(n)}}{2} \equiv U_{(j-\frac{1}{2})}^{(n)} \tag{3.64a}$$

$$\frac{U_{(j)}^{(n)} - U_{(j-1)}^{(n)}}{h_{(j)}} = \frac{V_{(j)}^{(n)} + V_{(j-1)}^{(n)}}{2} \equiv V_{(j-\frac{1}{2})}^{(n)} \tag{3.64b}$$

$$\begin{aligned}&\frac{V_{(j)}^{(n-\frac{1}{2})} - V_{(j-1)}^{(n-\frac{1}{2})}}{h_{(j)}} + (FV)_{(j-\frac{1}{2})}^{(n-\frac{1}{2})} + \\ &+\beta^{(n-\frac{1}{2})}\left[1 - \left(U^2\right)_{(j-\frac{1}{2})}^{(n-\frac{1}{2})}\right] = 2\xi^{(n-\frac{1}{2})}\left[U_{(j-\frac{1}{2})}^{(n-\frac{1}{2})}\frac{U_{(j-\frac{1}{2})}^{(n)} - U_{(j-\frac{1}{2})}^{(n-1)}}{k^{(n)}} - V_{(j-\frac{1}{2})}^{(n-\frac{1}{2})}\frac{F_{(j-\frac{1}{2})}^{(n)} - F_{(j-\frac{1}{2})}^{(n-1)}}{k^{(n)}}\right].\end{aligned}$$

Note that equations (3.64a) and (3.64b) are centered at $(\xi^n, \eta_{(j-\frac{1}{2})})$ whereas the third equation is centered at the midpoint $(\xi^{(n-\frac{1}{2})}, \eta_{(j-\frac{1}{2})})$ because also derivatives in streamwise direction are involved. Again, the last equation sorted such that current positions (terms including n) will be on the left-hand side, while terms that contain only previous positions or other terms that are known are arranged to the right. It follows

$$\begin{aligned}&\frac{V_{(j)}^{(n)} - V_{(j-1)}^{(n)}}{h_{(j)}} + (FV)_{(j-\frac{1}{2})}^{(n)} - \beta^{(n-\frac{1}{2})}\left(U^2\right)_{(j-\frac{1}{2})}^{(n)} - \\ &\quad - 2\frac{\xi^{(n-\frac{1}{2})}}{k^{(n)}}\left[\left(U^2\right)_{(j-\frac{1}{2})}^{(n)} - (FV)_{(j-\frac{1}{2})}^{(n)} - F_{(j-\frac{1}{2})}^{(n)}V_{(j-\frac{1}{2})}^{(n-1)} + F_{(j-\frac{1}{2})}^{(n-1)}V_{(j-\frac{1}{2})}^{(n)}\right] = R_{(j-\frac{1}{2})}^{(n-1)}\end{aligned} \tag{3.64c}$$

where the known terms and terms of the previous position are summarized in

$$R_{(j-\frac{1}{2})}^{(n-1)} = -L_{(j-\frac{1}{2})}^{(n-1)} + 2\frac{\xi^{(n-\frac{1}{2})}}{k^{(n)}}\left[-\left(U^2\right)_{(j-\frac{1}{2})}^{(n-1)} + (FV)_{(j-\frac{1}{2})}^{(n-1)}\right] \tag{3.65a}$$

$$L_{(j-\frac{1}{2})}^{(n-1)} = \frac{V_{(j)}^{(n-1)} - V_{(j-1)}^{(n-1)}}{h_{(j)}} + (FV)_{(j-\frac{1}{2})}^{(n-1)} + \beta^{(n-\frac{1}{2})}\left[2 - \left(U^2\right)_{(j-\frac{1}{2})}^{(n-1)}\right]. \tag{3.65b}$$

In a next step the Newton iterates are introduced to solve equations 3.64. The terms F, U and V are approximated as

$$F_{(j)}^{(n),(i+1)} = F_{(j)}^{(n),(i)} + \delta F_{(j)}^{(n),(i)}, \tag{3.66a}$$

$$U_{(j)}^{(n),(i+1)} = U_{(j)}^{(n),(i)} + \delta U_{(j)}^{(n),(i)}, \tag{3.66b}$$

$$V_{(j)}^{(n),(i+1)} = V_{(j)}^{(n),(i)} + \delta V_{(j)}^{(n),(i)} \tag{3.66c}$$

where i is the iteration number. As the boundary-layer equations are nonlinear the system has to be linearized which is done by neglecting all quadratic terms of $\delta F_{(j)}^{(n),(i)}$, $\delta U_{(j)}^{(n),(i)}$ and $\delta V_{(j)}^{(n),(i)}$. In the following the superscript n for an unknown variable will be dropped due to reason of simplicity. Then, the system of equations (3.64) expands to

$$F_{(j)}^{(i)} + \delta F_{(j)}^{(i)} - F_{(j-1)}^{(i)} - \delta F_{(j-1)}^{(i)} = \frac{h_{(j)}}{2}\left(U_{(j)}^{(i)} + \delta U_{(j)}^{(i)} + U_{(j-1)}^{(i)} + \delta U_{(j-1)}^{(i)}\right)$$

$$U_{(j)}^{(i)} + \delta U_{(j)}^{(i)} - U_{(j-1)}^{(i)} - \delta U_{(j-1)}^{(i)} = \frac{h_{(j)}}{2}\left(V_{(j)}^{(i)} + \delta V_{(j)}^{(i)} + V_{(j-1)}^{(i)} + \delta V_{(j-1)}^{(i)}\right)$$

$$\begin{aligned}
&\frac{V_{(j)}^{(i)} - V_{(j-1)}^{(i)}}{h_{(j)}} + (FV)_{(j-\frac{1}{2})}^{(i)} - \beta^{(n-\frac{1}{2})}\left(U^2\right)_{(j-\frac{1}{2})}^{(i)} - \\
&\quad - 2\frac{\xi^{(n-\frac{1}{2})}}{k^{(n)}}\left[\left(U^2\right)_{(j-\frac{1}{2})}^{(n)} - (FV)_{(j-\frac{1}{2})}^{(i)} - F_{(j-\frac{1}{2})}^{(i)} V_{(j-\frac{1}{2})}^{(n-1)} + F_{(j-\frac{1}{2})}^{(n-1)} V_{(j-\frac{1}{2})}^{(i)}\right] + \frac{\delta V_{(j)}^{(i)} - \delta V_{(j-1)}^{(i)}}{h_{(j)}} + \\
&\quad + \frac{1}{2}\left(F_{(j-\frac{1}{2})}^{(i)} \delta V_{(j)}^{(i)} + F_{(j-\frac{1}{2})}^{(i)} \delta V_{(j-1)}^{(i)} + V_{(j-\frac{1}{2})}^{(i)} \delta F_{(j)}^{(i)} + V_{(j-\frac{1}{2})}^{(i)} \delta F_{(j-1)}^{(i)}\right) - \\
&\quad - \beta^{(n-\frac{1}{2})}\left(U_{(j-\frac{1}{2})}^{(i)} \delta U_{(j)}^{(i)} + U_{(j-\frac{1}{2})}^{(i)} \delta U_{(j-1)}^{(i)}\right) - 2\frac{\xi^{(n-\frac{1}{2})}}{k^{(n)}}\left[U_{(j-\frac{1}{2})}^{(i)} \delta U_{(j)}^{(i)} + U_{(j-\frac{1}{2})}^{(i)} \delta U_{(j-1)}^{(i)} - \right. \\
&\qquad \left. - \frac{1}{2}\left(F_{(j-\frac{1}{2})}^{(i)} \delta V_{(j)}^{(i)} + F_{(j-\frac{1}{2})}^{(i)} \delta V_{(j)}^{(i-1)} + V_{(j-\frac{1}{2})}^{(i)} \delta F_{(j)}^{(i)} + V_{(j-\frac{1}{2})}^{(i)} \delta F_{(j)}^{(i-1)} + \right.\right. \\
&\qquad\quad \left.\left. + V_{(j-\frac{1}{2})}^{(n-1)} \delta F_{(j)}^{(i)} + V_{(j-\frac{1}{2})}^{(n-1)} \delta F_{(j-1)}^{(i)} - F_{(j-\frac{1}{2})}^{(n-1)} \delta V_{(j)}^{(i)} - F_{(j-\frac{1}{2})}^{(n-1)} \delta V_{(j-1)}^{(i)}\right)\right] = R_{(j-\frac{1}{2})}^{(n-1)}
\end{aligned}$$

which can be summarized as

$$\delta F_{(j)}^{(i)} - \delta F_{(j-1)}^{(i)} - \frac{h_{(j)}}{2}\left(\delta U_{(j)}^{(i)} + \delta U_{(j-1)}^{(i)}\right) = \left(r_1\right)_{(j)} \tag{3.67a}$$

$$\delta U_{(j)}^{(i)} - \delta U_{(j-1)}^{(i)} - \frac{h_{(j)}}{2}\left(\delta V_{(j)}^{(i)} + \delta V_{(j-1)}^{(i)}\right) = \left(r_3\right)_{(j-1)} \tag{3.67b}$$

$$\begin{aligned}
&\left(s_1\right)_{(j)} \delta V_{(j)}^{(i)} + \left(s_2\right)_{(j)} \delta V_{(j-1)}^{(i)} + \left(s_3\right)_{(j)} \delta F_{(j)}^{(i)} + \\
&\quad + \left(s_4\right)_{(j)} \delta F_{(j-1)}^{(i)} + \left(s_5\right)_{(j)} \delta U_{(j)}^{(i)} + \left(s_6\right)_{(j)} \delta U_{(j-1)}^{(i)} = \left(r_2\right)_{(j)}
\end{aligned} \tag{3.67c}$$

with

$$(r_1)_{(j)} = -\left(F^{(i)}_{(j)} - F^{(i)}_{(j-1)}\right) + h_{(j)} U^{(i)}_{(j-\frac{1}{2})} \tag{3.68a}$$

$$(r_3)_{(j-1)} = -\left(U^{(i)}_{(j)} - U^{(i)}_{(j-1)}\right) + h_{(j)} V^{(i)}_{(j-\frac{1}{2})} \tag{3.68b}$$

$$(r_2)_{(j)} = R^{(n-1)}_{(j-\frac{1}{2})} - \left[\frac{V^{(i)}_{(j)} - V^{(i)}_{(j-1)}}{h_{(j)}} + \alpha_1^{(n)} (FV)^{(i)}_{(j-\frac{1}{2})} - \alpha_2^{(n)} \left(U^2\right)^{(n)}_{(j-\frac{1}{2})} + \right. \\ \left. + \alpha^{(n)} \left(F^{(i)}_{(j-\frac{1}{2})} V^{(n-1)}_{(j-\frac{1}{2})} - F^{(n-1)}_{(j-\frac{1}{2})} V^{(i)}_{(j-\frac{1}{2})} \right) \right] \tag{3.68c}$$

and

$$(s_1)_{(j)} = \frac{1}{h_{(j)}} + \frac{\alpha_1^{(n)}}{2} F^{(i)}_{(j-\frac{1}{2})} - \frac{\alpha^{(n)}}{2} F^{(n-1)}_{(j-\frac{1}{2})} \tag{3.69a}$$

$$(s_2)_{(j)} = -\frac{1}{h_{(j)}} + \frac{\alpha_1^{(n)}}{2} F^{(i)}_{(j-\frac{1}{2})} - \frac{\alpha^{(n)}}{2} F^{(n-1)}_{(j-\frac{1}{2})} \tag{3.69b}$$

$$(s_3)_{(j)} = \frac{\alpha_1^{(n)}}{2} V^{(i)}_{(j-\frac{1}{2})} + \frac{\alpha^{(n)}}{2} V^{(n-1)}_{(j-\frac{1}{2})} \tag{3.69c}$$

$$(s_4)_{(j)} = \frac{\alpha_1^{(n)}}{2} V^{(i)}_{(j-\frac{1}{2})} + \frac{\alpha^{(n)}}{2} V^{(n-1)}_{(j-\frac{1}{2})} \tag{3.69d}$$

$$(s_5)_{(j)} = -\alpha_2^{(n)} U^{(i)}_{(j-\frac{1}{2})} \tag{3.69e}$$

$$(s_6)_{(j)} = -\alpha_2^{(n)} U^{(i)}_{(j-\frac{1}{2})}. \tag{3.69f}$$

The new variables introduced here are the constants

$$\alpha^{(n)} = 2\frac{\xi^{(n-\frac{1}{2})}}{k^{(n)}} \tag{3.70a}$$

$$\alpha_1^{(n)} = 1 + \alpha^{(n)} \tag{3.70b}$$

$$\alpha_2^{(n)} = \beta^{(n-\frac{1}{2})} + \alpha^{(n)}. \tag{3.70c}$$

Note that (3.67) to (3.70) are set up in the same manner as the system for the Falkner-Skan solution. Therefore, the numerical solution can be implemented in the same way. Again, the system has $3J + 3$ variables and $3J$ equations plus the three boundary conditions that result in

$$\delta F^{(i)}_{(0)} = 0 \tag{3.71a}$$

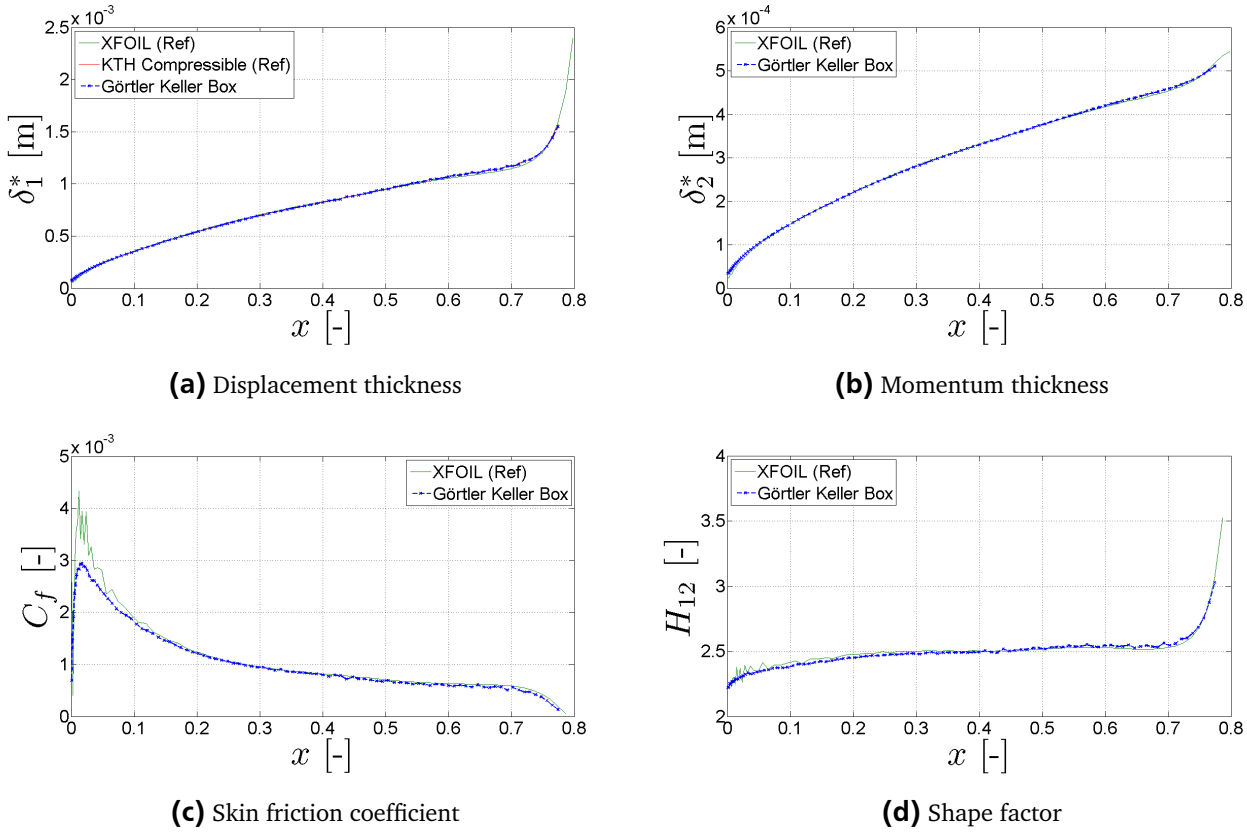
$$\delta U^{(i)}_{(0)} = 0 \tag{3.71b}$$

$$\delta U^{(i)}_{(J)} = 0. \tag{3.71c}$$

In order to implement an effective numerical scheme the system (3.67) is written in matrix form

$$\boldsymbol{A}^{(i)}\boldsymbol{\delta}^{(i)} = \boldsymbol{r}^{(i)} \tag{3.72}$$

which is block tridiagonal form as well. In *MATLAB* the system will be solved by defining the matrix $\boldsymbol{A}^{(i)}$ to be sparse and the internal *MATLAB* routine calls the Thomas algorithm as described in section 3.1.4. The block structure of system (3.72) is shown in equation (3.60) and each component of $\boldsymbol{A}^{(i)}$ is listed in (3.61).



(a) Displacement thickness

(b) Momentum thickness

(c) Skin friction coefficient

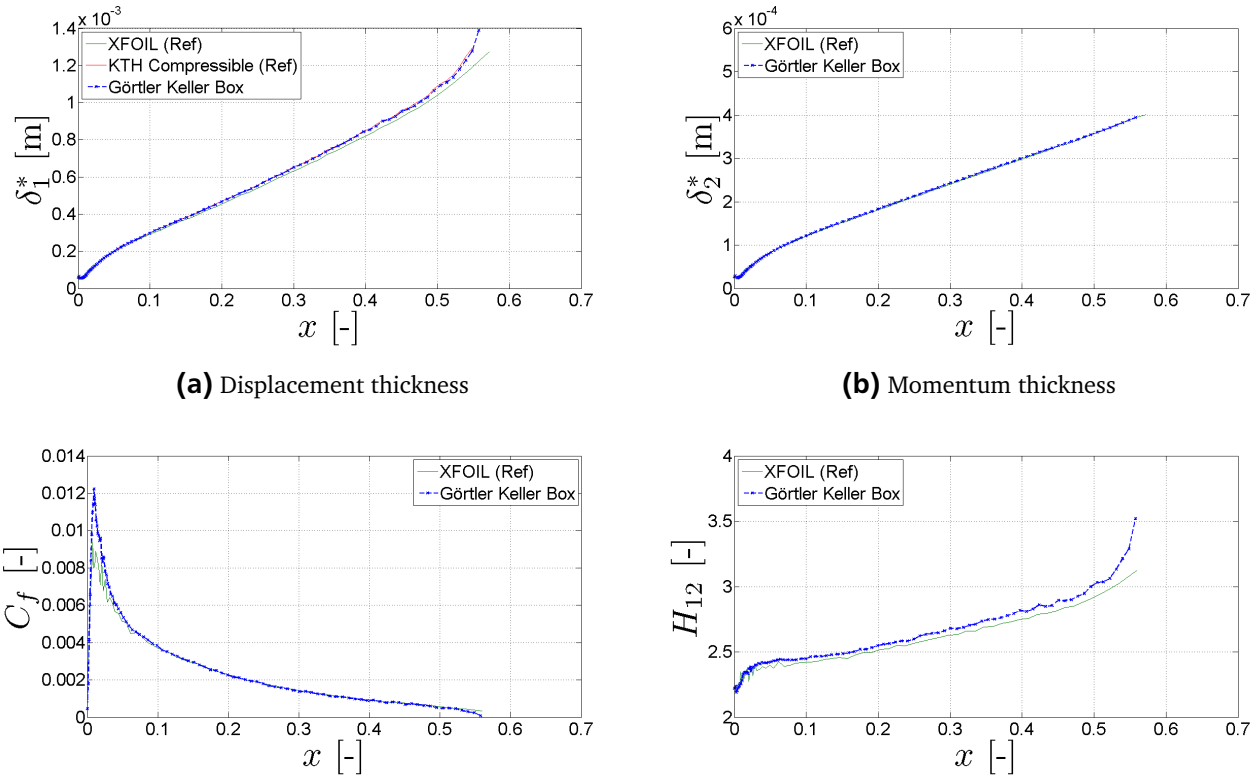
(d) Shape factor

Figure 3.19.: Boundary-layer characteristics of the pressure side for the Görtler transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

The solution at each streamwise position is used to calculate the profiles and their derivatives with equation (2.96) as well as the boundary-layer characteristics. In the following the results for a calculation of the *MW-airfoil* is shown. The angle of attack for the calculation is $\alpha = 3.3°$ for a Reynolds number of $Re = 2\,351\,842$ and a Mach number of $Ma = 0.01$. The converged solutions for the pressure and the suction side are depicted in Figures 3.19 to 3.22. The streamwise position where the profiles are shown is for the pressure side $x = 0.0584$ which corresponds to $x_c = 0.0594$ in airfoil coordinates and for the suction side it is $x = 0.0689$ and $x_c = 0.0373$, respectively.

In Figure 3.19 the boundary-layer characteristics for the pressure side are illustrated. The profiles match qualitatively very well with the reference data from *XFOIL*. Slight discrepancies in the absolute values can be traced back to the fact that the pressure distribution for the *XFOIL* data is slightly different. *XFOIL* uses an interactive method to solve the profile and thus the pressure distribution is slightly changed to

converge. The onset of transition is detected as shown in Figure 3.19d because the shape factor increase is similar to the *XFOIL* data. Further, the numerical scheme is very stable as shown in Figures 3.19c and 3.19d where the vibrations at the beginning almost completely vanish. The different height of the peak in Figure 3.19c can be explained with the different pressure distribution.
The boundary-layer characteristics for the suction side are illustrated in Figure 3.20. Again the qualitative behavior is detected very well when compared to the converged solution of *XFOIL*. As shown in Figure 3.20a the displacement thickness perfectly matches with the reference data from the KTH solver which uses the same pressure distribution. As shown in Figures 3.20c and 3.20d the numerical scheme is stable and the vibrations of the *XFOIL* data vanishes. However, the smoothness decreases for positions that approach the separation point as shown in Figure 3.20a. Further, the onset of transition is computed earlier as shown in Figure 3.20d because the shape factor rises earlier. The decrease of the displacement and momentum thickness at the beginning in Figures 3.20a and 3.20b refers to the stagnation point till the leading edge. In this region the profiles become "fuller" resulting in a decrease. After the leading edge the flow accelerates and the values increase again.



(a) Displacement thickness

(b) Momentum thickness

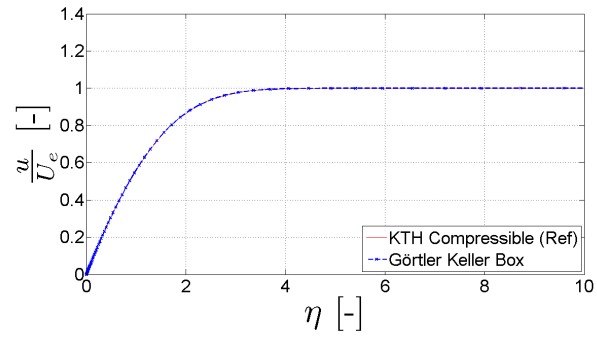
(c) Skin friction coefficient

(d) Shape factor

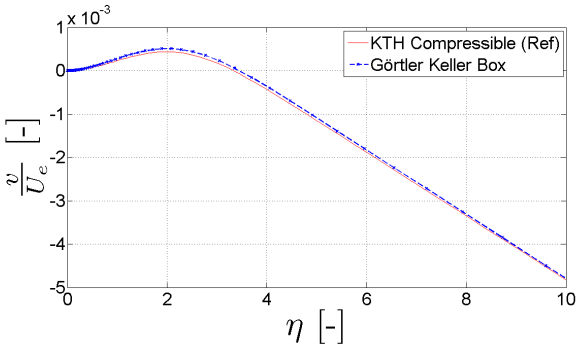
Figure 3.20.: Boundary-layer characteristics of the suction side for the Görtler transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

The profiles for the pressure side are depicted in Figure 3.21 for a position of $x = 0.0584$ which corresponds to $x_c = 0.0594$. The profiles for the streamwise velocity component and its derivatives in wall-normal direction match perfectly with the reference data from the KTH solver as shown in Figures 3.21a, 3.21c and 3.21e. The qualitative behavior of streamwise derivative of u (Figure 3.21g) is detected very well even though the absolute values slightly differ at the peak near the wall and the asymptotic value. In general, the wall-normal velocity component is computed with larger discrepancies than the streamwise component. The discrepancies rise with increase derivatives as shown in Figures 3.21d and

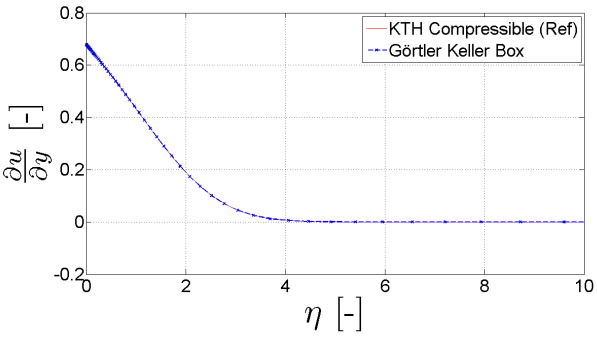
3.21f. Those differences could be explained with the different discretization methods as the edge velocity might be computed slightly different. Thereby, the shape slightly changes and the profiles which have a low order of magnitude do not match perfectly with the reference data. The streamwise derivative of v as shown in Figure 3.21b is computed qualitatively and quantitatively wrong which could be explained with the same reason.
In Figure 3.22 the boundary-layer profiles for the suction side are depicted for a position of $x = 0.0689$ which corresponds to $x_c = 0.0373$. Likewise for the pressure side the velocity profiles of the streamwise component u match very well with the KTH reference data. The small bump near the wall in Figure 3.22e can be explained with the different discretization. The velocity profiles are computed for a location very close to the leading edge and the resulting differences in the pressure distribution can be relatively large. The qualitative behavior for the streamwise derivative as shown in Figure 3.22g is detected very well and the absolute values are match quite good with the reference data. Again, the profiles for the wall-normal component v show larger discrepancies to the references data. Even though the trend is detected quite well, especially the second derivative in wall-normal direction shows a wrong value at the wall (Figure 3.22f). The streamwise derivative in Figure 3.22h shows qualitatively and quantitatively a wrong shape.



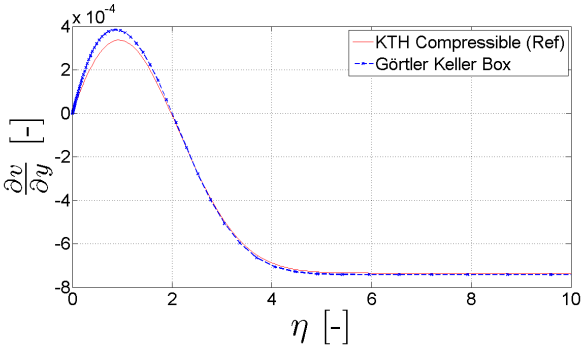
(a) Streamwise velocity u



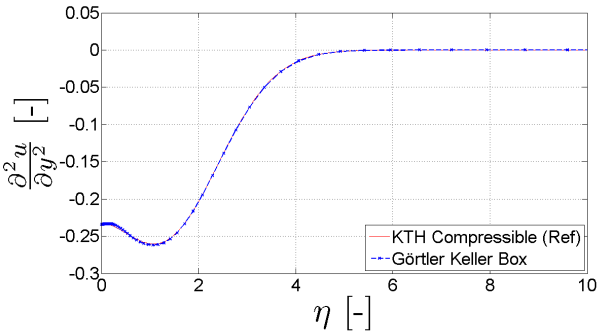
(b) Wall-normal velocity v



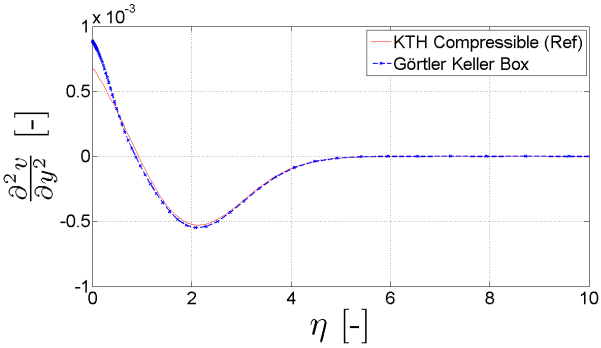
(c) Wall-normal derivative of u



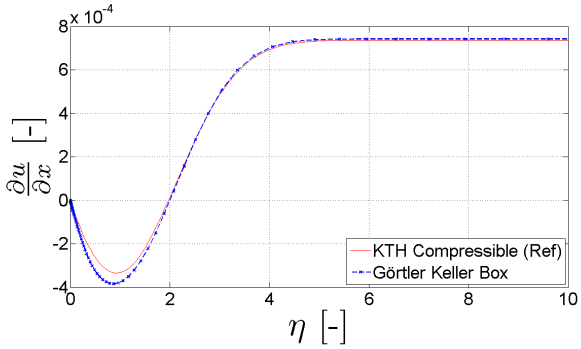
(d) Wall-normal derivative of v



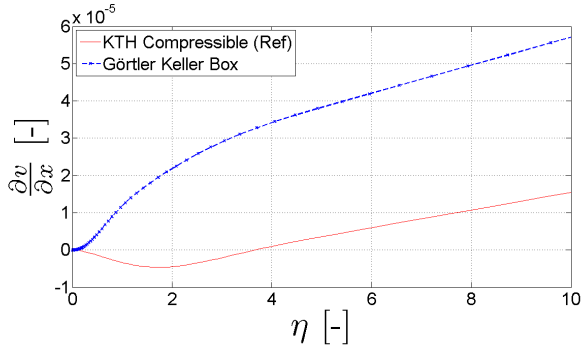
(e) Second wall-normal derivative of u



(f) Second wall-normal derivative of v

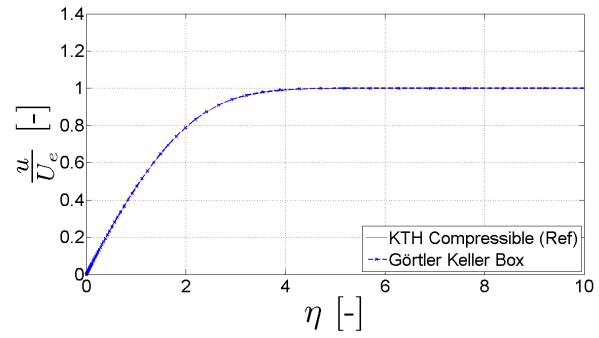


(g) Streamwise derivative of u

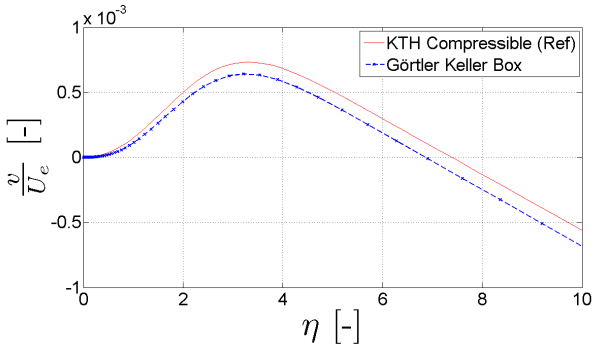


(h) Streamwise derivative of v

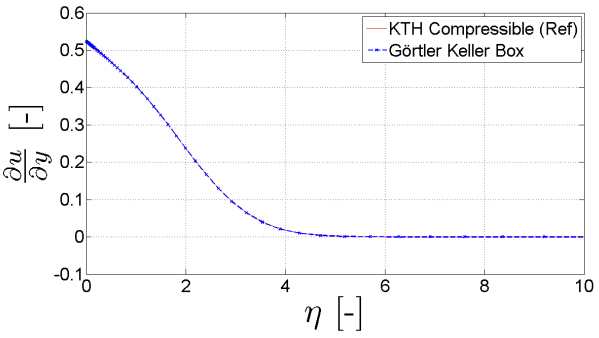
Figure 3.21.: Boundary-layer profiles of the pressure side at $x_c = 0.0594$ for the Görtler transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).



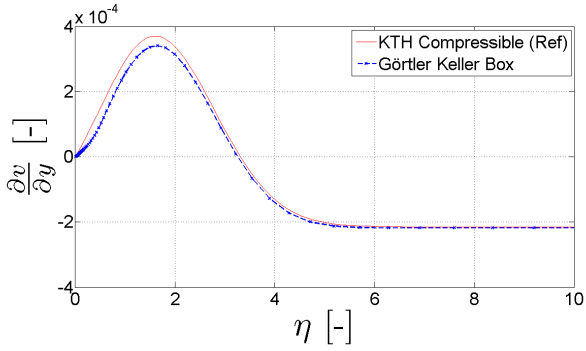
(a) Streamwise velocity u



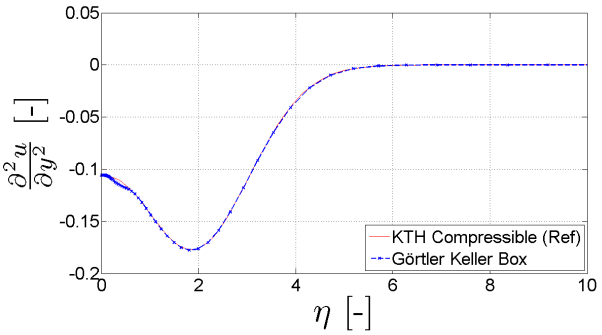
(b) Wall-normal velocity v



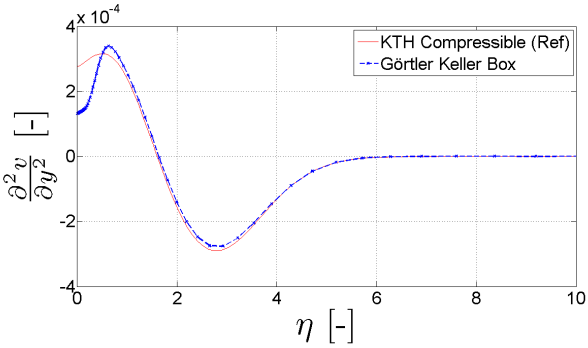
(c) Wall-normal derivative of u



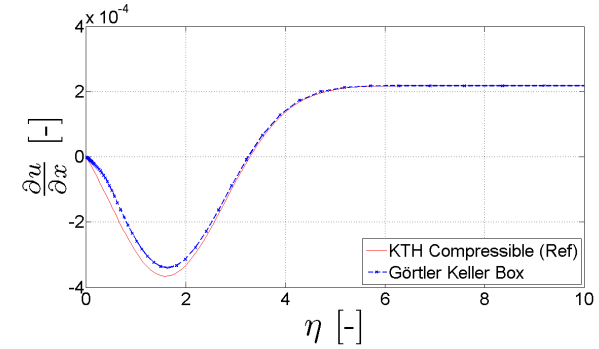
(d) Wall-normal derivative of v



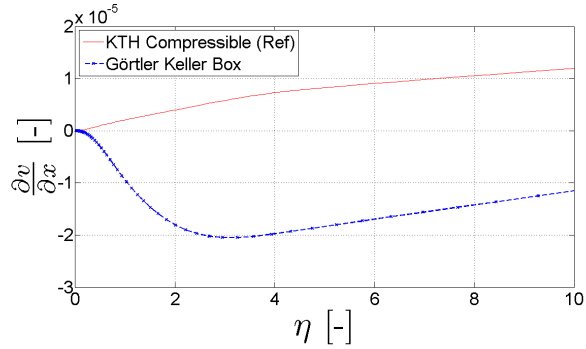
(e) Second wall-normal derivative of u



(f) Second wall-normal derivative of v



(g) Streamwise derivative of u



(h) Streamwise derivative of v

Figure 3.22.: Boundary-layer profiles of the suction side at $x_c = 0.0373$ for the Görtler transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

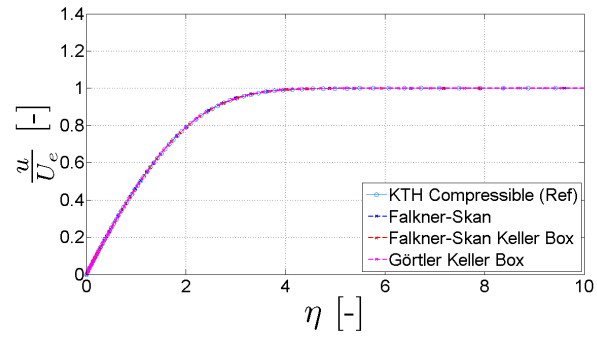
3.4.3 Comparison of Boundary-Layer Calculation Methods

In sections 3.4.1 and 3.4.2 two different discretization methods and three different results are presented. In the following a short evaluation of the different methods is given and the main focus will be on the boundary-layer profiles as they are needed for a stability analysis. The boundary-layer characteristics are very similar for all computations. The profiles for the suction side show a larger discrepancy as for the pressure side even though the findings are qualitatively the same. Therefore, the evaluation is simply based on the profiles of the suction side. The case of a flow with a Reynolds number of $Re = 2\,351\,842$ and a Mach number of $Ma = 0.01$ under an angle of attack of 3.3°. The position for the stagnation point at which the profiles are compared is $x = 0.0689$ which corresponds to $x_c = 0.0373$. The stagnation point is on the bottom of the profile and thus the flow has to go around the leading edge before acceleration. The chosen position thus shows profiles that start to develop and thus significant differences are observed. Further, it should be remarked that the compressible boundary-layer solver of KTH is also based on a Falkner-Skan transformation and the implementation is very similar to the direct method shown in section 3.4.1.

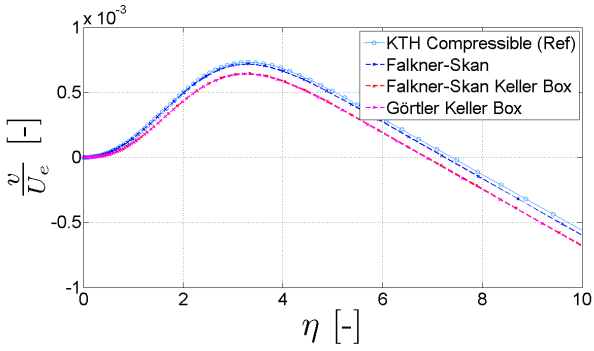
In Figure 3.23 the profiles for the suction side are plotted for the three different methods described above. As already mentioned before the qualitative behavior for the profiles is quite good except in Figure 3.23h where only the direct discretization is in close agreement with the KTH solver. In general, the results of the methods using the Keller box are very similar and only slight differences are noticed for the second derivatives in wall-normal direction at the wall as shown in Figures 3.23e and 3.23f. The streamwise derivative of v in Figure 3.23h depicts completely different trends which could be explained by the different scaling of the streamwise coordinate.

In general, the direct discretization shows very good agreement with the reference data for all plots. The reason for that is the very similar implementation. As no other reference data is available the profiles of the direct method should be used for transition prediction via stability analysis. However, other reference data may be needed to draw a final conclusion. Maybe DNS data can be used to evaluate the profiles and it might turn out that the Keller box gives better results. Moreover, the direct discretization results in a well-structured code in which alterations can easily be implemented. The solution directly results in the boundary-layer profiles and thus no complicated calculations to find the profiles have to be performed.

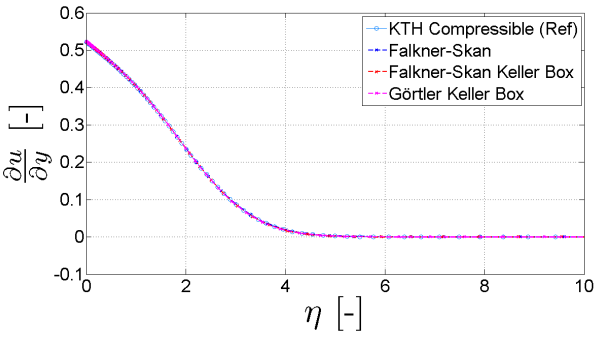
In contrast to that the discretization with the Keller box is numerically more complex as four points are used for discretization. Further, the solution gives the dimensionless stream function and its derivatives which then has to be transformed to find the boundary-layer profiles and the derivatives. The differences to reference data of the KTH solver are larger than for the direct discretization but other reference data should be used to draw a final conclusion on that. The benefit of the Keller box discretization is the easy way to alter the calculation to an inverse method as presented in section 4.2. Further, the Görtler transformation is introduced. For this transformation the wall-normal variable is not supposed to grow and this could be quite useful when a turbulence model is included. In this work the turbulence model has only be implemented for the direct and Keller box discretization with the Falkner-Skan transformation which will be presented in section 4.



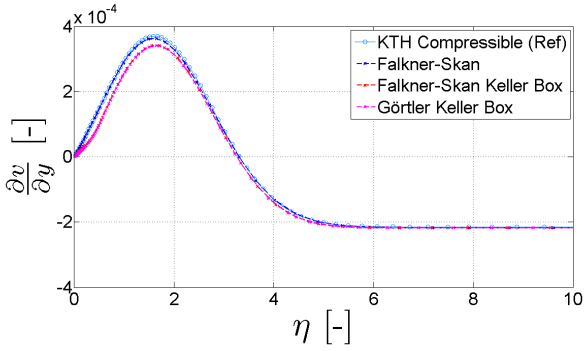
(a) Streamwise velocity u



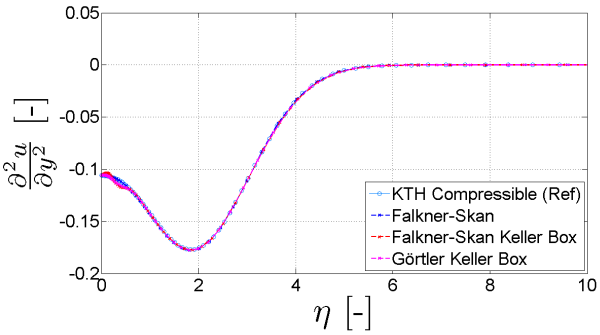
(b) Wall-normal velocity v



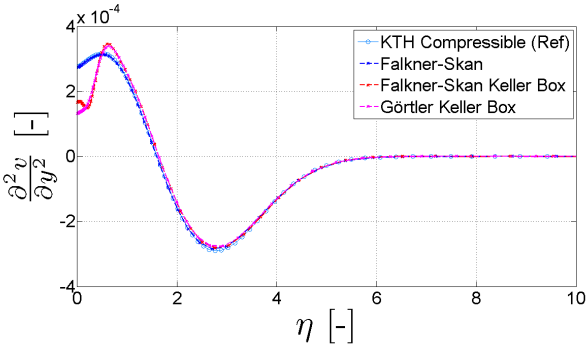
(c) Wall-normal derivative of u



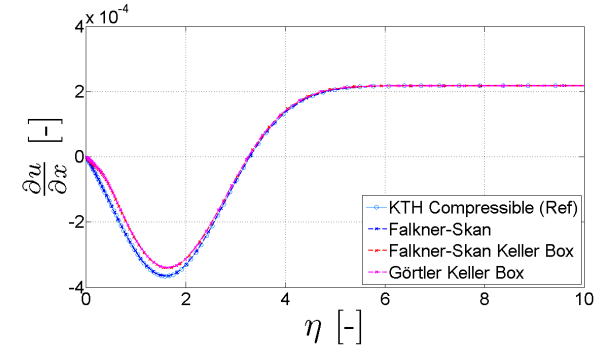
(d) Wall-normal derivative of v



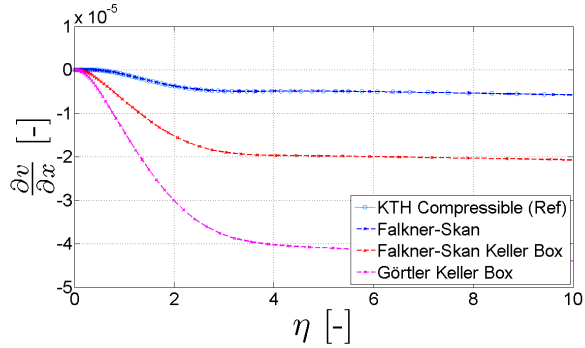
(e) Second wall-normal derivative of u



(f) Second wall-normal derivative of v



(g) Streamwise derivative of u



(h) Streamwise derivative of v

Figure 3.23.: Comparison of the boundary-layer profiles of the suction side at $x_c = 0.0373$ (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

4 Implementation of a Turbulence Model and an Inverse Boundary-Layer Method

Whereas the previous chapter dealt with laminar flows and flows without separation, in the following chapter more complicated flows are investigated. Therefore, the turbulence model described in section 2.6 is implemented. In the second part an inverse method is implemented that also iterates the edge velocity. This method is used to calculate airfoil flows with separation. The relevant equations are derived in section 2.7.

4.1 Cebeci and Smith Turbulence Model

In the following section the turbulence model of Cebeci and and Smith will be applied to the direct discretization of the boundary-layer equations and to the discretization using the Keller box. The implementation for each method will be described shortly. As the turbulent viscosity ν_t is calculated in the same way for each method the procedure will be outlined first.
The inner turbulent viscosity $\nu_{t,i}$ is computed in the following way. At first, the friction velocity

$$u_\tau = \sqrt{\frac{U_e}{Re} f_{\eta\eta}\Big|_{\eta=0}} \sqrt{Re\frac{U_e}{x}} \tag{4.1}$$

is computed. After that the dimensionless pressure

$$p^+ = \frac{U_e}{Reu_\tau^3}\frac{dU_e}{dx} \tag{4.2}$$

can be found. Inserting those values into equation (2.108) results in N which is used to calculate

$$A = \frac{26}{ReNu_\tau}\frac{U_e}{Reu_\tau^3}\frac{dU_e}{dx}. \tag{4.3}$$

The parameter A represents a damping length constant. In combination with the dimensionless wall-normal distance y as given in equation (2.71), the mixing length is found to be

$$l = \kappa y\left[1 - e^{-\frac{y}{A}}\right]. \tag{4.4}$$

In order to prevent numerical problems the absolute value of the mixing length is used. The reason for these problems is that the dimensionless pressure p^+ can become complex. The last parameter that is independent of the quantities above is the intermittency factor from equation (2.110). The intermittency factor does not change for the streamwise position and thus can be computed outside of the iteration loop. Finally, the inner part of the turbulent viscosity is found to be

$$\nu_{t,i} = l^2\left|U_e f_{\eta\eta}\sqrt{Re\frac{U_e}{x}}\right|\gamma_{tr}. \tag{4.5}$$

The turbulent viscosity for the outer region is calculated next. The value of the displacement thickness is computed by equation (2.89). The boundary-layer thickness δ is defined by 99.5% of the edge velocity as defined in [8]. The physical value of the wall-normal distance can be replaced with η, the dimensionless distance such that

$$\gamma = \left[1 + 5.5\left(\frac{\eta}{\eta_{99.5}}\right)^6\right]^{-1}. \tag{4.6}$$

The outer part of the turbulent viscosity is then found to be

$$\nu_{t,o} = \alpha U_e \delta_1 \gamma_{tr} \gamma. \tag{4.7}$$

Finally, the inner and outer turbulent viscosities are compared. If the outer viscosity becomes smaller than the inner viscosity the values of the outer viscosity are taken. In the following implementation the ratio of the turbulent and the physical viscosity in dimensional values is used which is found to be

$$\frac{\nu_t^*}{\nu^*} = \nu_t Re. \tag{4.8}$$

In the following, the Cebeci and Smith turbulence model will be implemented for the direct discretization and the Keller box discretization. In both cases the Falkner-Skan transformation is used to transform the wall-normal coordinate.

4.1.1 Direct Discretization Method

Before the implementation is shown the new boundary-layer equation including the turbulent viscosity is derived. It is found to be

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = U_e\frac{\partial U_e}{\partial x} + \frac{1}{Re}\left(1 + \frac{\nu_t^*}{\nu^*}\right)\frac{\partial^2 u}{\partial y^2} + \frac{1}{Re}\frac{\partial \frac{\nu_t^*}{\nu^*}}{\partial y}\frac{\partial u}{\partial y}. \tag{4.9}$$

The discretization of the boundary-layer equation with turbulence model then becomes

$$\left(a + b\frac{\partial}{\partial \eta} + c\frac{\partial^2}{\partial \eta^2}\right)u^{(i)} = U_e\frac{\partial U_e}{\partial x} - u^{(i)}\left(D_{x,n-1}u^{(n-1)} + D_{x,n-2}u^{(n-2)}\right) \tag{4.10}$$

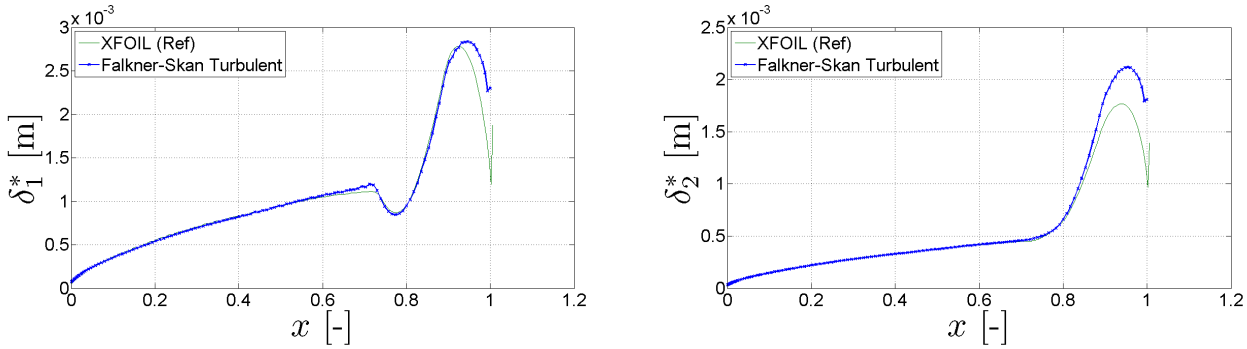
with

$$a = D_{x,n}u^{(i)} \tag{4.11a}$$

$$b = u^{(i)}\frac{\partial \eta}{\partial x} + v^{(i)}\frac{\partial \eta}{\partial y} - \frac{1}{Re}\frac{\partial\left(\frac{\nu_t^*}{\nu^*}\right)^{(i)}}{\partial \eta}\left(\frac{\partial \eta}{\partial y}\right)^2 \tag{4.11b}$$

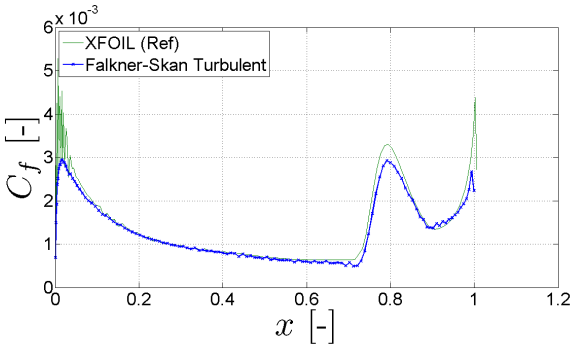
$$c = -\frac{1}{Re}\left[1 + \left(\frac{\nu_t^*}{\nu^*}\right)^{(i)}\right]\left(\frac{\partial \eta}{\partial y}\right)^2. \tag{4.11c}$$

The term ν_t^*/ν^* is the output of the turbulence model described at the beginning of the section. The

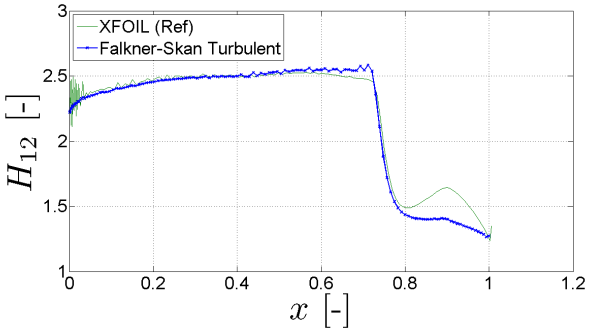


(a) Displacement thickness

(b) Momentum thickness



(c) Skin friction coefficient



(d) Shape factor

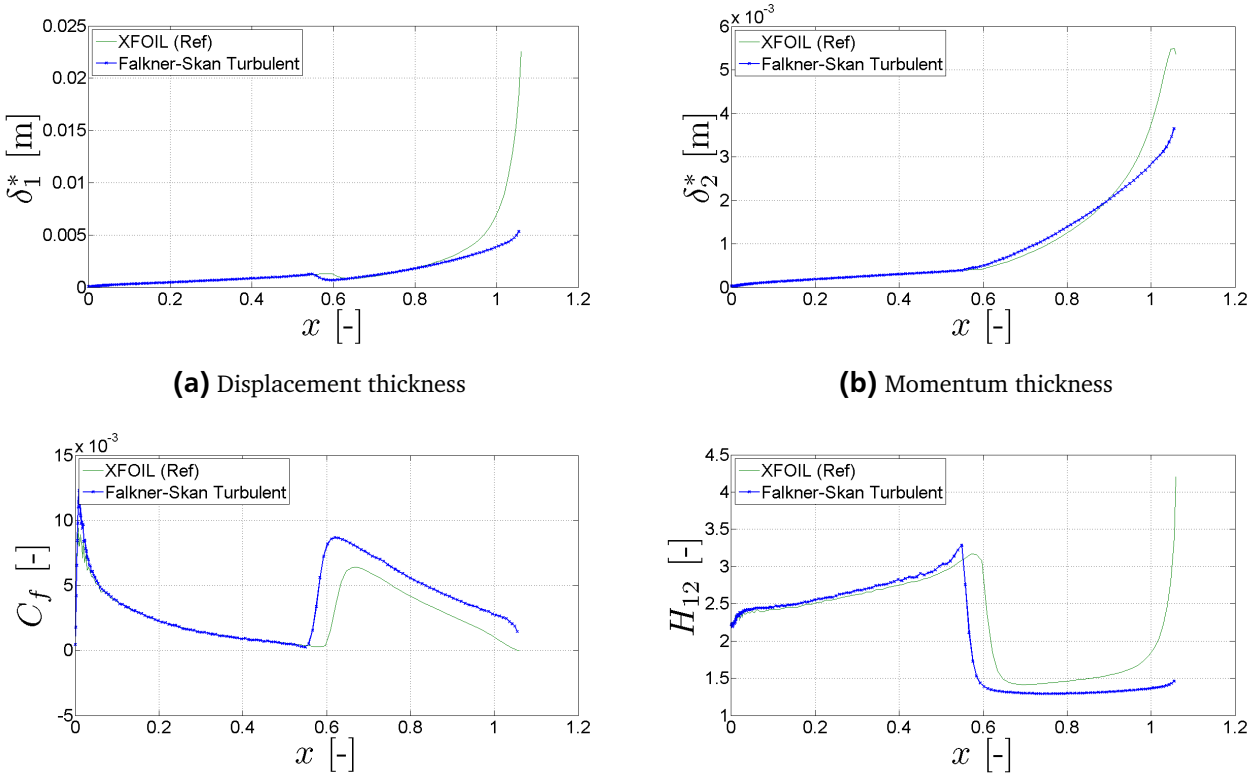
Figure 4.1.: Boundary-layer characteristics of the pressure side for the Falkner-Skan transformation with turbulence model (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

solution of equation (4.10) is described in section 3.4.1. First the system is solved for the streamwise component u and then the continuity equation is used to integrate the wall-normal velocity v. Convergence is achieved if the wall-normal derivative of u at the first grid point off the wall is

$$\frac{\frac{\partial u_{(1)}^{(i)}}{\partial \eta} - \frac{\partial u_{(1)}^{(i-1)}}{\partial \eta}}{\frac{\partial u_{(1)}^{(i-1)}}{\partial \eta}} < 10^{-10}. \tag{4.12}$$

The calculation is then repeated for the next streamwise position up to the trailing edge.
In Figure 4.1 the resulting boundary-layer characteristics for the pressure side are shown. Note that the transition point is set at $x_{tr} = 0.72$ manually. The transition is also set manually in *XFOIL* in order to avoid the transition prediction for a separation bubble which occurs if it is not set manually. The results for the positions before the transition point match exactly with the characteristics shown in section 3.4.1 for the strictly laminar calculation. However, the first thing to notice is that the computation does not stop at the point where laminar separation would take place. This is because of the turbulence model which changes the right-hand side. The behavior for the displacement thickness and momentum thickness in Figure 4.1a and 4.1b is qualitatively correct and also the quantitative values match with *XFOIL* quite well. The height of the peak for the momentum thickness is predicted slightly higher as for the reference value. The computation for the shape factor and the friction coefficient agrees also very well with the reference as shown in Figure 4.1d and 4.1c. The oscillations at the beginning do not occur which suggests stable numerics. The smaller prediction of the second peak in the shape factor results from the higher peak

prediction in the momentum thickness. The small differences to the reference probably result from the slightly change pressure distribution. As already mentioned earlier the *XFOIL* calculation interactively fits the edge velocity during the calculation and thus changes the distribution. The second reason for those differences could be the implemented turbulence models. Probably, a different model is implemented in *XFOIL* that leads to those different results.
The boundary-layer characteristics for the suction side are shown in Figure 4.2 and the transition location is set manually at $x_{tr} = 0.55$. The overall agreement for the suction side is much better than for the pressure side. The qualitative behavior is matched for all characteristics and also the quantitative differences are smaller. The early transition cannot be avoided as the method does not converge for a laminar flow for values larger than the transition position. In the following the shape is exactly met if one adds the distance between the transition point of the reference and the manually set transition. However, the large increase in displacement and momentum thickness can not be accounted for by the turbulence model. The possible reason is a different turbulence model used by the reference.



(a) Displacement thickness

(b) Momentum thickness

(c) Skin friction coefficient

(d) Shape factor

Figure 4.2.: Boundary-layer characteristics of the suction side for the Falkner-Skan transformation with turbulence model (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

4.1.2 Keller Box Discretization Method

The derivation of the turbulent boundary layer for the Keller box is a little different compared to the direct derivation shown above. The total viscosity will be used for the calculation and a separation into the physical and the turbulent component of the viscosity is avoided. This total viscosity is

$$b = 1 + \frac{\nu_t^*}{\nu^*}, \tag{4.13}$$

where the fraction is computed in the turbulent model as shown in section 4.1. For laminar flows $b = 1$ as the turbulent part is zero.
By applying the discretization of the Keller box only the boundary-layer equation (3.51c) changes to

$$\begin{aligned}\frac{(bv)_{(j)}^{(n)} - (bv)_{(j-1)}^{(n)}}{h_{(j)}} + \frac{m^{(n-\frac{1}{2})}+1}{2}\left(fv\right)_{(j-\frac{1}{2})}^{(n)} - m^{(n-\frac{1}{2})}\left(u^2\right)_{(j-\frac{1}{2})}^{(n)} - \\ - \frac{x^{(n-\frac{1}{2})}}{k^{(n)}}\left[\left(u^2\right)_{(j-\frac{1}{2})}^{(n)} - \left(fv\right)_{(j-\frac{1}{2})}^{(n)} - f_{(j-\frac{1}{2})}^{(n)} v_{(j-\frac{1}{2})}^{(n-1)} + f_{(j-\frac{1}{2})}^{(n-1)} v_{(j-\frac{1}{2})}^{(n)}\right] = R_{(j-\frac{1}{2})}^{(n-1)}\end{aligned} \tag{4.14}$$

and the term $L_{(j-\frac{1}{2})}^{(n-1)}$ of $R_{(j-\frac{1}{2})}^{(n-1)}$ becomes

$$L_{(j-\frac{1}{2})}^{(n-1)} = \frac{(bv)_{(j)}^{(n-1)} - (bv)_{(j-1)}^{(n-1)}}{h_{(j)}} + \frac{m^{(n-\frac{1}{2})}+1}{2}\left(fv\right)_{(j-\frac{1}{2})}^{(n-1)} + m^{(n-\frac{1}{2})}\left[2 - \left(u^2\right)_{(j-\frac{1}{2})}^{(n-1)}\right]. \tag{4.15}$$

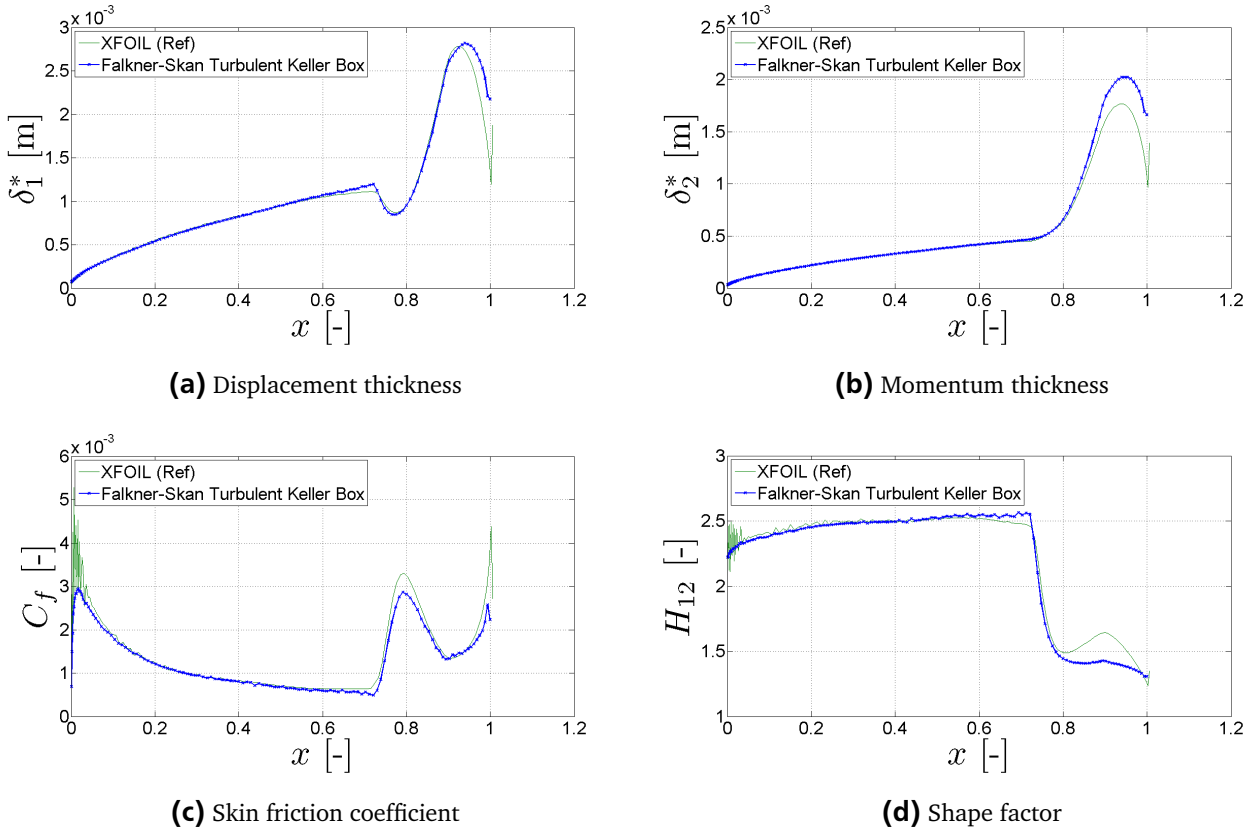
This means that simply the total viscosity b is added before the wall-normal derivative of $f_{\eta\eta}$. The following simplifications are similar to the one shown in section 3.4.2 and thus will not be repeated here. The only terms that change are $\left(s_1\right)_{(j)}$, $\left(s_1\right)_{(j)}$ and $\left(r_2\right)_{(j)}$ which now read

$$\left(s_1\right)_{(j)} = \frac{b_{(j)}^{(i)}}{h_{(j)}} + \frac{\alpha_1^{(n)}}{2} f_{(j-\frac{1}{2})}^{(i)} - \frac{\alpha^{(n)}}{2} f_{(j-\frac{1}{2})}^{(n-1)} \tag{4.16a}$$

$$\left(s_2\right)_{(j)} = -\frac{b_{(j-1)}^{(i)}}{h_{(j)}} + \frac{\alpha_1^{(n)}}{2} f_{(j-\frac{1}{2})}^{(i)} - \frac{\alpha^{(n)}}{2} f_{(j-\frac{1}{2})}^{(n-1)} \tag{4.16b}$$

$$\begin{aligned}\left(r_2\right)_{(j)} = R_{(j-\frac{1}{2})}^{(n-1)} - \Bigg[\frac{(bv)_{(j)}^{(i)} - (bv)_{(j-1)}^{(i)}}{h_{(j)}} + \alpha_1^{(n)}\left(fv\right)_{(j-\frac{1}{2})}^{(i)} - \alpha_2^{(n)}\left(u^2\right)_{(j-\frac{1}{2})}^{(n)} + \\ + \alpha^{(n)}\left(f_{(j-\frac{1}{2})}^{(i)} v_{(j-\frac{1}{2})}^{(n-1)} - f_{(j-\frac{1}{2})}^{(n-1)} v_{(j-\frac{1}{2})}^{(i)}\right)\Bigg].\end{aligned} \tag{4.16c}$$

The equations can then be transfered into a block diagonal matrix system and solved as shown before. The setup of that system does not change and thus the solution procedure has not to be amended. The only component that is added is the update of the total viscosity b in each iteration step.
In Figure 4.3 the resulting boundary-layer characteristics for the pressure side are shown. Note that the transition point is manually set at $x_{tr} = 0.72$. It is also manually set in *XFOIL* to avoid the prediction with the build-in laminar separation prediction as already mentioned before. The results for the positions



(a) Displacement thickness

(b) Momentum thickness

(c) Skin friction coefficient

(d) Shape factor

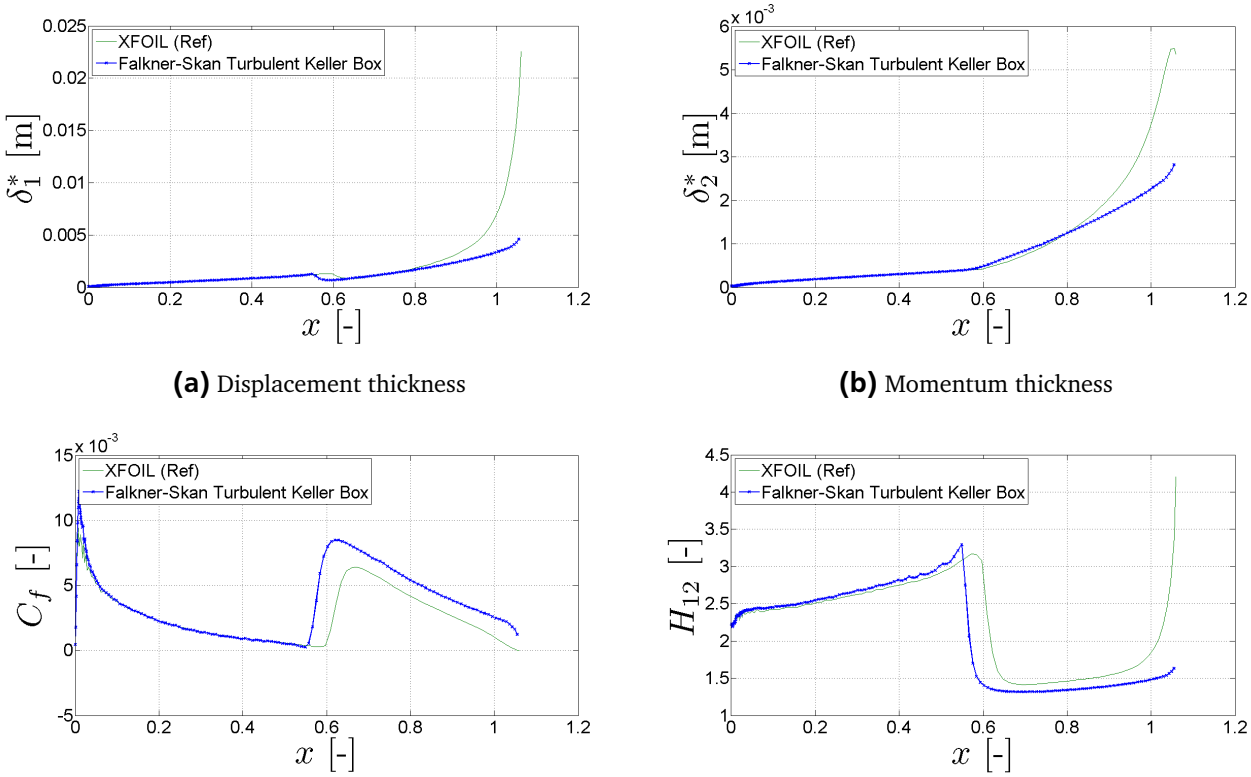
Figure 4.3.: Boundary-layer characteristics of the pressure side for the Falkner-Skan transformation with turbulence model using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

before the transition point match exactly with the characteristics shown in section 3.4.2 for the strictly laminar calculation. Like for the direct discretization, the first thing to notice is that the calculation does not stop at the separation point. This is because of the turbulence model which changes the right hand side. The results match very well with the reference and the shapes are very similar to the direct discretization. Again, oscillations in the skin friction and shape factor plots in Figures 4.3c and 4.3d are reduced resulting in very stable numerics. The slightly different shapes in the turbulent part of the plots leads to the conclusion that *XFOIL* uses obviously a different turbulence model which thus makes different predictions.

The boundary-layer characteristics for the suction side are depicted in Figure 4.4 and the transition location is set manually at $x_{tr} = 0.55$. The overall agreement for the suction side is much better than for the pressure side. The same observations as in above's section for the direct discretization apply. Again the results match perfectly with those shown in Figure 4.2.

The implementation of the Cebeci and Smith turbulence model for the Görtler transformation should be carried out in the future as well. As already mentioned in section 2.5.2 the Görtler transformation considers the growth of the boundary layer in streamwise direction. Especially in a turbulent flow the growth of the boundary layer can be quite large and results could be different for the Görtler transformation.

Nevertheless, a final conclusion about the turbulence model cannot be drawn yet as other reference data should be used to evaluate the results. A calculation using a RANS solver could be carried out or even LES/DNS data for other airfoil flows could be used to evaluate the results of the Cebeci and Smith turbulence model more detailed.



(a) Displacement thickness

(b) Momentum thickness

(c) Skin friction coefficient

(d) Shape factor

Figure 4.4.: Boundary-layer characteristics of the suction side for the Falkner-Skan transformation with turbulence model using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

4.2 Implementation of the Inverse Boundary Layer

In the last sections the standard boundary-layer method was implemented and also it was shown how to include a turbulence model. In the following the possibility to switch to an inverse method is applied. This is needed when the boundary-layer flow encounters separation and the edge velocity cannot be prescribed as a boundary condition anymore.
The perturbation of the edge velocity $\delta U_e(x)$ given in equation (2.118) can be discretized as [8]

$$\delta U_e(x) = \sum_{j=1}^{i-1} C_{ij} D_j + C_{ii} D_i + \sum_{j=i+1}^{i-1} C_{ij} D_j \tag{4.17}$$

where the coefficients are found to be [5]

$$C_{ij} = \frac{1}{\pi}\left(E^i_j - E^i_{j+1}\right) \quad \text{with} \quad E^i_1 = E^i_{J+1} = 0 \tag{4.18a}$$

$$D = U_e \delta_1 = \sqrt{\frac{x}{Re}}\left(Y_{max} U_e - F(x, Y_{max})\right) \tag{4.18b}$$

and

$$E^i_j = \frac{1}{x^{(n)} - x^{(n-1)}} \ln\left|\frac{x^{(i)} - x^{(n-1)}}{x^{(i)} - x^{(n)}}\right| \quad \text{for} \quad j \neq i \text{ and } j \neq i+1 \tag{4.19a}$$

$$E^i_j = \frac{\frac{x^{(i+1)} - x^{(i)}}{x^{(i+1)} - x^{(i-1)}} \ln\left|\frac{x^{(i)} - x^{(i-1)}}{x^{(i)} - x^{(i+1)}}\right| + 2}{x^{(i)} - x^{(i-1)}} \tag{4.19b}$$

$$E^i_{j+1} = \frac{\frac{x^{(i)} - x^{(i-1)}}{x^{(i+1)} - x^{(i-1)}} \ln\left|\frac{x^{(i)} - x^{(i-1)}}{x^{(i)} - x^{(i+1)}}\right| - 2}{x^{(i+1)} - x^{(i)}}. \tag{4.19c}$$

Finally, by employing the discretization of $\delta U_e(x)$ the boundary condition at the free-stream can also be written as

$$Y = Y_{max}: \quad F_Y - C_{ii} D_i = U^0_e + \sum_{j=1}^{i-1} C_{ij} D_j + \sum_{j=i+1}^{i-1} C_{ij} D_j = g_i. \tag{4.20}$$

In summary, all boundary conditions read

$$\begin{aligned} Y = 0: &\qquad F = 0 \quad \text{and} \quad F_Y = 0 \\ Y = Y_{max}: &\qquad F_Y = U_e(x) \quad \text{and} \quad \lambda F + (1 - \lambda Y) U_e. \end{aligned} \tag{4.21}$$

where $\lambda = C_{ii}\sqrt{x/Re}$.
By combining the standard and the inverse method the following boundary-layer equation can be found which can also include a turbulence model

$$\left(b f_{\eta\eta}\right)_\eta + \lambda_1 \left[\frac{m}{2} f f_{\eta\eta} + m\left(1 - f_\eta\right)^2\right] + \frac{1}{2} f f_{\eta\eta} = x\left(f_\eta f_{x\eta} - f_{\eta\eta} f_x - \lambda_2 U_e \frac{dU_e}{dx}\right) \tag{4.22}$$

where $\lambda_1 = 1$ and $\lambda_2 = 0$ for the standard method and $\lambda_1 = 0$ and $\lambda_2 = 1$ for the inverse method. The parameters λ_1 and λ_2 do not refer to the dimensionless pressure parameter used for the integral

boundary-layer equations in section 2.4. In contrast to the standard method the edge velocity in the inverse mode is also an unknown that has to be iterated. Note that the transformation also changes when switching from the inverse to the standard method or vice versa. Further, the change of U_e does not depend on η and thus $U_{e\eta} = 0$. By transforming equation (4.22) into three differential equations of first order it follows

$$\frac{U_{e(j)}^{(n)} - U_{e(j-1)}^{(n)}}{h_{(j)}} = 0 \tag{4.23a}$$

$$\frac{f_{(j)}^{(n)} - f_{(j-1)}^{(n)}}{h_{(j)}} = \frac{u_{(j)}^{(n)} + u_{(j-1)}^{(n)}}{2} \equiv u_{(j-\frac{1}{2})}^{(n)} \tag{4.23b}$$

$$\frac{u_{(j)}^{(n)} - u_{(j-1)}^{(n)}}{h_{(j)}} = \frac{v_{(j)}^{(n)} + v_{(j-1)}^{(n)}}{2} \equiv v_{(j-\frac{1}{2})}^{(n)} \tag{4.23c}$$

$$\begin{aligned} &\frac{(bv)_{(j)}^{(n)} - (bv)_{(j-1)}^{(n)}}{h_{(j)}} + \lambda_1 \frac{m^{(n-\frac{1}{2})}}{2} \left[(fv)_{(j-\frac{1}{2})}^{(n)} - 2\left(u^2\right)_{(j-\frac{1}{2})}^{(n)} \right] + \frac{1}{2}(fv)_{(j-\frac{1}{2})}^{(n)} - \\ &- \frac{x^{(n-\frac{1}{2})}}{k^{(n)}} \left[\left(u^2\right)_{(j-\frac{1}{2})}^{(n)} - (fv)_{(j-\frac{1}{2})}^{(n)} - f_{(j-\frac{1}{2})}^{(n)} v_{(j-\frac{1}{2})}^{(n-1)} + f_{(j-\frac{1}{2})}^{(n-1)} v_{(j-\frac{1}{2})}^{(n)} - \lambda_2 \left(U_e^2\right)_{(j-\frac{1}{2})}^{(n)} \right] = R_{(j-\frac{1}{2})}^{(n-1)} \end{aligned} \tag{4.23d}$$

with

$$R_{(j-\frac{1}{2})}^{(n-1)} = -L_{(j-\frac{1}{2})}^{(n-1)} + \frac{x^{(n-\frac{1}{2})}}{k^{(n)}} \left[-\left(u^2\right)_{(j-\frac{1}{2})}^{(n-1)} + (fv)_{(j-\frac{1}{2})}^{(n-1)} + \lambda_2 \left(U_e^2\right)_{(j-\frac{1}{2})}^{(n-1)} \right] \tag{4.23e}$$

$$L_{(j-\frac{1}{2})}^{(n-1)} = \frac{v_{(j)}^{(n-1)} - v_{(j-1)}^{(n-1)}}{h_{(j)}} + \lambda_1 \frac{m^{(n-\frac{1}{2})}}{2} \left[(fv)_{(j-\frac{1}{2})}^{(n-1)} + 4 - 2\left(u^2\right)_{(j-\frac{1}{2})}^{(n-1)} \right] + \frac{1}{2}(fv)_{(j-\frac{1}{2})}^{(n-1)}. \tag{4.23f}$$

In the next step the unknowns are approximated by adding small perturbations for f, u, v and U_e as described in section 3.4.2. After some simplifications the system reads

$$\delta U_{e(j)}^{(i)} - \delta U_{e(j-1)}^{(i)} = (r_4)_{(j-1)} \tag{4.24a}$$

$$\delta f_{(j)}^{(i)} - \delta f_{(j-1)}^{(i)} - \frac{h_{(j)}}{2}\left(\delta u_{(j)}^{(i)} + \delta u_{(j-1)}^{(i)}\right) = (r_1)_{(j)} \tag{4.24b}$$

$$\delta u_{(j)}^{(i)} - \delta u_{(j-1)}^{(i)} - \frac{h_{(j)}}{2}\left(\delta v_{(j)}^{(i)} + \delta v_{(j-1)}^{(i)}\right) = (r_3)_{(j-1)} \tag{4.24c}$$

$$\begin{aligned} &(s_1)_{(j)}\,\delta v_{(j)}^{(i)} + (s_2)_{(j)}\,\delta v_{(j-1)}^{(i)} + (s_3)_{(j)}\,\delta f_{(j)}^{(i)} + (s_4)_{(j)}\,\delta f_{(j-1)}^{(i)} + \\ &+ (s_5)_{(j)}\,\delta u_{(j)}^{(i)} + (s_6)_{(j)}\,\delta u_{(j-1)}^{(i)} + (s_7)_{(j)}\,\delta U_{e(j)}^{(i)} + (s_8)_{(j)}\,\delta U_{e(j-1)}^{(i)} = (r_2)_{(j)} \end{aligned} \tag{4.24d}$$

with

$$(r_4)_{(j-1)} = -\left(U_{e(j)}^{(i)} - U_{e(j-1)}^{(i)}\right) \tag{4.25a}$$

$$(r_1)_{(j)} = -\left(f_{(j)}^{(i)} - f_{(j-1)}^{(i)}\right) + h_{(j)} u_{(j-\frac{1}{2})}^{(i)} \tag{4.25b}$$

$$(r_3)_{(j-1)} = -\left(u_{(j)}^{(i)} - u_{(j-1)}^{(i)}\right) + h_{(j)} v_{(j-\frac{1}{2})}^{(i)} \tag{4.25c}$$

$$(r_2)_{(j)} = R_{(j-\frac{1}{2})}^{(n-1)} - \left[\frac{(bv)_{(j)}^{(i)} - (bv)_{(j-1)}^{(i)}}{h_{(j)}} + \alpha_1^{(n)} (fv)_{(j-\frac{1}{2})}^{(i)} - \alpha_2^{(n)} \left(u^2\right)_{(j-\frac{1}{2})}^{(n)} + \right. \\ \left. + \alpha^{(n)} \left(f_{(j-\frac{1}{2})}^{(i)} v_{(j-\frac{1}{2})}^{(n-1)} - f_{(j-\frac{1}{2})}^{(n-1)} v_{(j-\frac{1}{2})}^{(i)} + \lambda_2 \left(U_e^2\right)_{(j-\frac{1}{2})}^{(i)}\right)\right] \tag{4.25d}$$

and

$$(s_1)_{(j)} = \frac{b_{(j)}^{(i)}}{h_{(j)}} + \frac{\alpha_1^{(n)}}{2} f_{(j-\frac{1}{2})}^{(i)} - \frac{\alpha^{(n)}}{2} f_{(j-\frac{1}{2})}^{(n-1)} \tag{4.26a}$$

$$(s_2)_{(j)} = -\frac{b_{(j-1)}^{(i)}}{h_{(j)}} + \frac{\alpha_1^{(n)}}{2} f_{(j-\frac{1}{2})}^{(i)} - \frac{\alpha^{(n)}}{2} f_{(j-\frac{1}{2})}^{(n-1)} \tag{4.26b}$$

$$(s_3)_{(j)} = \frac{\alpha_1^{(n)}}{2} v_{(j-\frac{1}{2})}^{(i)} + \frac{\alpha^{(n)}}{2} v_{(j-\frac{1}{2})}^{(n-1)} \tag{4.26c}$$

$$(s_4)_{(j)} = \frac{\alpha_1^{(n)}}{2} v_{(j-\frac{1}{2})}^{(i)} + \frac{\alpha^{(n)}}{2} v_{(j-\frac{1}{2})}^{(n-1)} \tag{4.26d}$$

$$(s_5)_{(j)} = -\alpha_2^{(n)} u_{(j-\frac{1}{2})}^{(i)} \tag{4.26e}$$

$$(s_6)_{(j)} = -\alpha_2^{(n)} u_{(j-\frac{1}{2})}^{(i)}. \tag{4.26f}$$

$$(s_7)_{(j)} = \lambda_2 \alpha^{(n)} U_{e_{(j-\frac{1}{2})}}^{(i)} \tag{4.26g}$$

$$(s_8)_{(j)} = \lambda_2 \alpha^{(n)} U_{e_{(j-\frac{1}{2})}}^{(i)}. \tag{4.26h}$$

The variables which are introduced here are the constants

$$\alpha^{(n)} = \frac{x^{(n-\frac{1}{2})}}{k^{(n)}} \tag{4.27a}$$

$$\alpha_1^{(n)} = \frac{1}{2} + \lambda_1 \frac{m^{(n-\frac{1}{2})}}{2} + \alpha^{(n)} \tag{4.27b}$$

$$\alpha_2^{(n)} = \lambda_1 m^{(n-\frac{1}{2})} + \alpha^{(n)}. \tag{4.27c}$$

The boundary conditions are implemented into the right-hand side as

$$(r_1)_{(0)} = 0 \tag{4.28a}$$

$$(r_2)_{(0)} = 0 \tag{4.28b}$$

$$(r_3)_{(J)} = \gamma_3 \tag{4.28c}$$

$$(r_4)_{(J)} = U_{e(J)}^{(i)} - u_{(J)}^{(i)} \tag{4.28d}$$

with

$$\text{Standard:}\quad \gamma_1 = 0 \qquad \text{Inverse:}\quad \gamma_1 = \lambda \tag{4.29a}$$

$$\text{Standard:}\quad \gamma_2 = 1 \qquad \text{Inverse:}\quad \gamma_2 = 1 - \lambda\eta_{(J)} \tag{4.29b}$$

$$\text{Standard:}\quad \gamma_3 = 0 \qquad \text{Inverse:}\quad \gamma_3 = g_i - \left(\gamma_1 f_{(J)}^{(i)} + \gamma_2 U_{e(J)}^{(i)}\right). \tag{4.29c}$$

The system is also written in block tridiagonal matrix form like in equation (3.60) with 4x4 matrices instead of 3x3 matrices for the standard problem to simplify the solution for *MATLAB*. The matrices then become

$$\boldsymbol{A}_{(0)}^{(i)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & -\frac{h_{(1)}}{2} & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \qquad \boldsymbol{A}_{(j)}^{(i)} = \begin{pmatrix} 1 & -\frac{h_{(j)}}{2} & 0 & 0 \\ (s_3)_{(j)} & (s_5)_{(j)} & (s_1)_{(j)} & (s_7)_{(j)} \\ 0 & -1 & -\frac{h_{(j+1)}}{2} & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \quad \text{for} \quad 1 \le j \le J-1$$

$$\boldsymbol{A}_{(J)}^{(i)} = \begin{pmatrix} 1 & -\frac{h_{(J)}}{2} & 0 & 0 \\ (s_3)_{(J)} & (s_5)_{(J)} & (s_1)_{(J)} & (s_7)_{(J)} \\ \gamma_1 & 0 & 0 & \gamma_2 \\ 0 & 1 & 0 & -1 \end{pmatrix} \tag{4.30a}$$

$$\boldsymbol{B}_{(j)}^{(i)} = \begin{pmatrix} -1 & -\frac{h_{(j)}}{2} & 0 & 0 \\ (s_4)_{(j)} & (s_6)_{(j)} & (s_2)_{(j)} & (s_7)_{(J)} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad \text{for} \quad 1 \le j \le J \tag{4.30b}$$

$$\boldsymbol{C}_{(j)}^{(i)} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & -\frac{h_{(j+1)}}{2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{for} \quad 0 \le j \le J-1 \tag{4.30c}$$

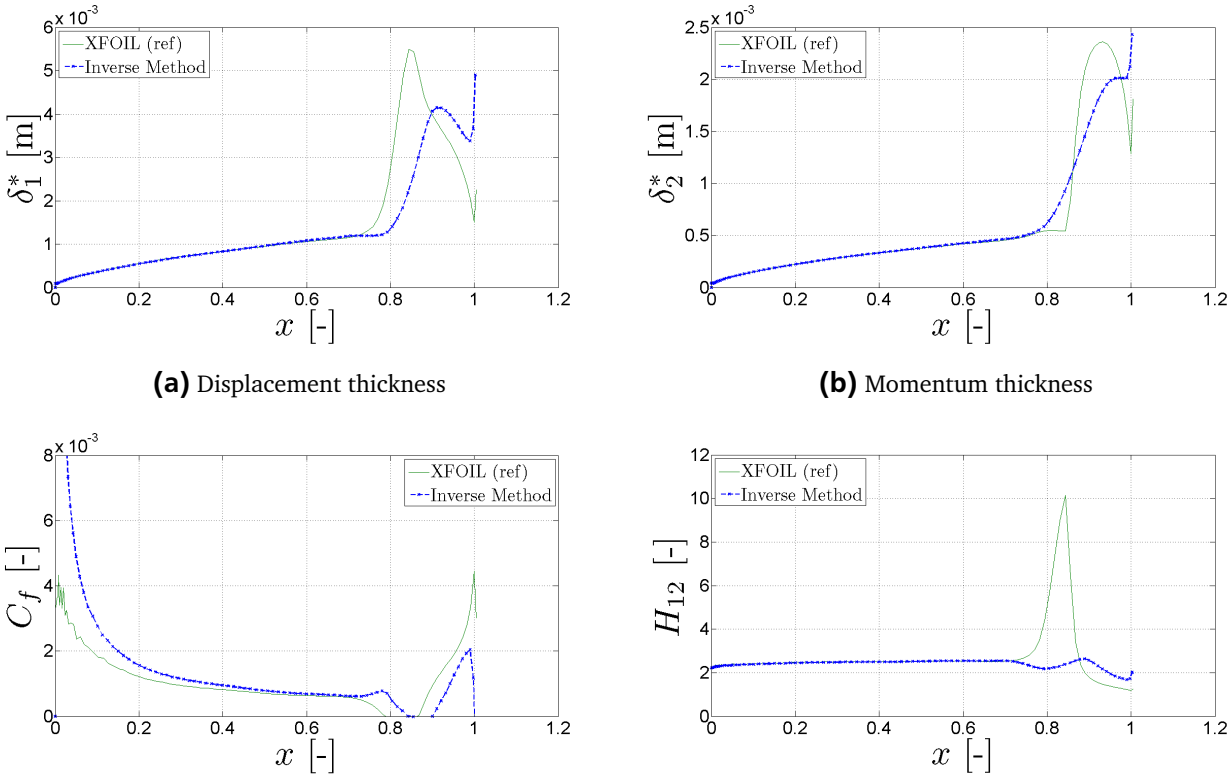
and

$$\vec{\delta}_{(j)}^{(i)} = \begin{pmatrix} \delta f_{(j)}^{(i)} \\ \delta u_{(j)}^{(i)} \\ \delta v_{(j)}^{(i)} \\ \delta U_{e(j)}^{(i)} \end{pmatrix} \quad \text{for} \quad 0 \le j \le J \tag{4.31a}$$

$$\vec{r}_{(j)}^{(i)} = \begin{pmatrix} (r_1)_{(j)} \\ (r_2)_{(j)} \\ (r_3)_{(j)} \\ (r_4)_{(j)} \end{pmatrix} \quad \text{for} \quad 0 \le j \le J. \tag{4.31b}$$

The system can be solved with a similar algorithm as described in section 3.1.4 but for 4x4 matrices. In

MATLAB such a routine is used to solve the sparse matrix $\boldsymbol{A}$.
In the following the boundary-layer characteristics are compared to the reference data from *XFOIL*. In the case of the inverse boundary-layer method the calculation is performed till the trailing edge as the computation does not stop at the separation point. The calculation is performed for the same case as compared before. The angle of attack for the calculation is $\alpha = 3.3°$ for a Reynolds number of $Re = 2\,351\,842$.
The boundary-layer characteristics for the pressure side are shown in Figure 4.5. The transition point is manually set to $x_{tr} = 0.7$ and thus the onset of turbulence might slightly differ from the reference data. The transition point marks the begin to calculate the viscosity by adding a turbulent part. It is not the position where the flow becomes turbulent. The displacement thickness and momentum thickness in Figures 4.5a and 4.5b agree with the reference data qualitatively. However, at the end of the profile large discrepancies can be found. On the one hand the onset of transition is starts later for the inverse method and on the other hand the peaks are computed quantitatively wrong. Both calculations depend on the turbulence model which obviously differs from the *XFOIL* turbulence model. Further, the flow builds a separation bubble close to the trailing edge as visible by the peak in the skin friction plot in Figure 4.5c. The separation bubble is larger in the *XFOIL* case which explains the high peak of the shape factor in Figure 4.5d. The separation occurs at the dent of the laminar profile which is needed to retain the momentum balance of the airfoil.



(a) Displacement thickness

(b) Momentum thickness

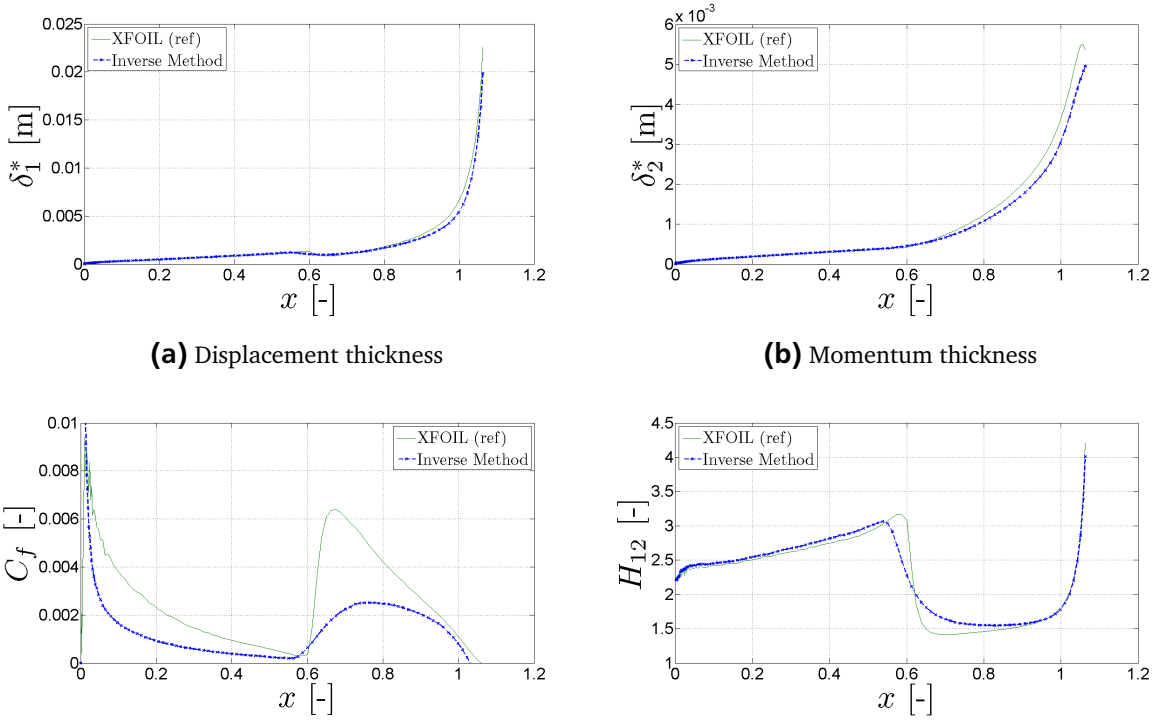
(c) Skin friction coefficient

(d) Shape factor

Figure 4.5.: Boundary-layer characteristics of the pressure side for the inverse method (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

In Figure 4.6 the boundary-layer characteristics for the suction side are shown. The calculations for the displacement thickness and the momentum thickness in Figure 4.6a and 4.6b match quite good with the results from *XFOIL*. Note that the transition point is set manually to $x_{tr} = 0.5$ and thus it might slightly

differs from the onset of turbulence in the reference data. The decrease of the displacement thickness before the onset of transition is not as sharp as in the *XFOIL* case. A plausible reason for that could be a different turbulence model in *XFOIL*. The rise of the displacement thickness for the turbulent region is qualitatively and quantitatively well met. The same conclusions can be drawn for the momentum thickness in Figure 4.5b. The early onset of transition and the smoother decrease in displacement thickness leads to the differences in the shape factor in Figure 4.6d. The largest discrepancies are found for the skin friction coefficient in Figure 4.6c. On the one hand is the peak near the stagnation point far overestimated. On the other hand is the peak after the transition point significantly underestimated. The first peak might be explained with some numerical difficulties close to the stagnation point but the exact problem could not be found yet. The different height of the second peak could be explained with a different turbulence model used in *XFOIL*. An implication of that would be a different shape of the velocity profile leading to a different derivative at the wall and thus to a different skin friction coefficient. Nevertheless, the friction coefficient was much better met for the standard method before and a good reason for the large discrepancies is not found yet.



(a) Displacement thickness

(b) Momentum thickness

(c) Skin friction coefficient

(d) Shape factor

Figure 4.6.: Boundary-layer characteristics of the suction side for the inverse method (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

5 Implementation of Stability Analysis for Transition Prediction

In the following chapter the implementation of the Orr-Sommerfeld and Squire solver for a local stability method is shown. The results are used to predict the transition point based on the local theory. Afterwards the implementation of the parabolized stability equations is presented where accounts for non-parallel effects. It is compared to the local stability solutions. First of all, the Chebyshev discretization is explained as it is implemented in both stability codes.

5.1 The Chebyshev Discretization

In order to discretize the linear stability equations (2.139) and (2.143) Chebyshev polynomials are introduced. The boundary-layer solver used finite differences for discretization instead. The advantage of Chebyshev polynomials is the higher accuracy. Spectral accuracy can be achieved. Chebyshev polynomials are a sequence of orthogonal polynomials that can be set up recursively. The important fact to notice is that the roots of those polynomials can be easily found. Thereby, a very accurate approximation of function can be given. The Chebyshev polynomials of the first kind are defined as [21]

$$T_0(\hat{y}) = 1 \tag{5.1a}$$

$$T_1(\hat{y}) = y \tag{5.1b}$$

$$T_{n+1}(\hat{y}) = 2\hat{y}\,T_n(\hat{y}) - T_{n-1}(\hat{y}) \tag{5.1c}$$

in the case of a recurrence relation. The domain $\hat{y}$ is $[-1, 1]$. The polynomial can also be calculated directly as [21]

$$T_n(\hat{y}) = \frac{1}{2}\left[\left(\hat{y} + \sqrt{\hat{y}^2 - 1}\right)^n + \left(\hat{y} - \sqrt{\hat{y}^2 - 1}\right)^n\right]. \tag{5.2}$$

Like the differential operator described in section 3.1.3, Chebyshev polynomials can also be used to build differentiation matrices. The computation of such matrices is efficiently implemented by employing the package of Weideman and Reddy [27]. However, the differentiation matrices are not sparse anymore and thus require higher numerical cost. The finite difference differentiation matrices were sparse and thus could be solved more easily.

The Chebyshev polynomials are used to set up a differentiation matrix for the wall-normal direction. These matrices are computed by the function `chebdif.m`. The input for the function `chebdif.m` is the maximum number of collocation points used for the calculation and the highest order needed for the calculation. The maximum number of collocation points depends on the size of the considered domain which must be large enough to fulfill the boundary conditions in the free-stream (2.134) but for the linear stability problem 100 collocation points should be enough. By utilizing the function `chebdif.m` the collocation points and the differentiation matrices are obtained. Note that the domain for Chebyshev

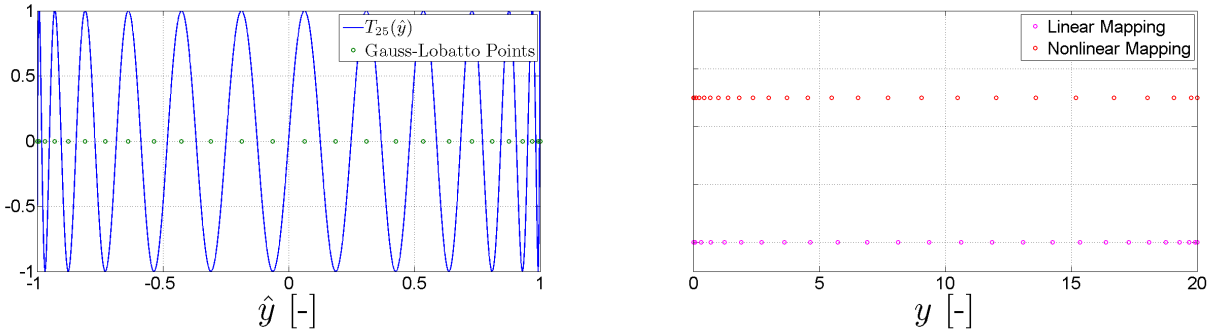
polynomials is $[1,-1]$ and that the first and last entry of `chebdif.m` are 1 and -1, respectively. The exact distribution is

$$\hat{y}_j = \cos\left(\frac{j\pi}{N}\right) \quad \text{for} \quad j = 0,1,\ldots,N. \tag{5.3}$$

where N is the total number of collocation points and those locations are called Gauss-Lobatto points [21]. In Figure 5.1a an example is depicted where $N = 25$. The resulting Chebyshev polynomial is shown in combination with the 26 extrema of that polynomial. Those extrema are the Gauss-Lobatto points. By increasing N the approximation becomes better.
In order to map the Chebyshev domain onto the computational domain that starts at 0 and ends at the upper limit of the domain, a mapping function has to be introduced that transforms the Gauss-Lobatto grid to the numerical grid. A linear mapping function can be given as

$$y_j = \frac{1}{2}\left[(b-a)\,\hat{y}_j + b - a\right] \tag{5.4}$$

where y_j is the physical grid point of $\hat{y}_j$ and a is the lower while b is the upper limit of the physical domain. In the case of the linear stability analysis $a = 0$ is the domain starts at the wall and $b = y_{max}$ representing the height of the box. A great advantage of employing Chebyshev polynomials is the distribution of collocation points as shown in Figure 5.1a. The collocation points are mostly located at the wall and the freestream. Thereby, the boundary conditions can be matched very accurately.



(a) Gauss-Lobatto points for 25th Chebyshev polynomial **(b)** Linear and Nonlinear mapping in physical domain

Figure 5.1.: Comparison between Chebyshev and physical domain for different mapping functions.

Using that transformation the physical grid points can be calculated and the derivative becomes

$$\frac{\partial \boldsymbol{u}}{\partial y} = \frac{\partial \hat{y}}{\partial y}\frac{\partial}{\partial \hat{y}}\boldsymbol{u} = \boldsymbol{M}\boldsymbol{D_1}\boldsymbol{u} \tag{5.5}$$

where $\boldsymbol{D_1}$ is the first order Chebyshev differentiation matrix and $\boldsymbol{M}$ is a diagonal matrix containing the elements $\frac{\partial \hat{y}_j}{\partial y_j} = \frac{2}{y_{max}}$ on its trace for the linear mapping function given in equation (5.4).
A different mapping function will be introduced as well which is closer resolved to the wall. Thereby, the Tollmien-Schlichting waves can be computed with more accuracy as changes are most significant close to the wall. The nonlinear function is given as

$$y_j = a\frac{1+\hat{y}_j}{b-\hat{y}_j} \tag{5.6}$$

with

$$a = \frac{y_i y_{max}}{y_{max} - 2y_i} \quad \text{and} \quad b = 1 + \frac{2a}{y_{max}} \tag{5.7}$$

where y_i is the location at which half of the grid points are located. In other words, by choosing a value close to the wall, half of the grid points are distributed within $0 \leq y \leq y_i$. The effect of this mapping is shown in Figure 5.1b. The transformation for the nonlinear mapping is $\frac{\partial \tilde{y}_j}{\partial y_j} = \frac{(b-\tilde{y})^2}{a(b+1)}$.
Furthermore, numerical integration can be performed as a matrix multiplication by setting up a Chebyshev integral weight function such that

$$\int_{-1}^{1} f(y)dy = \boldsymbol{W}\boldsymbol{f} \tag{5.8}$$

where $\boldsymbol{W}$ is the Chebyshev integral weight matrix which has diagonal form and $\boldsymbol{f}$ is the function that needs to be integrated. The weights read [21]

$$W(y_j) = \frac{1}{\frac{\partial \tilde{y}}{\partial y}} \frac{b_j}{N} \left\{ 2 + \sum_{n=2}^{N} c_n \frac{1+(-1)^n}{1-n^2} \cos\left(\frac{nj\pi}{N}\right) \right\}. \tag{5.9}$$

The first term is the partial derivative that is used to transform the Gauss-Lobatto grid into the physical domain. The constants are found to be

$$b_j = \begin{cases} 0.5, & \text{for } j = 0 \\ 1, & \text{for } j = 1, 2, \ldots, J-1 \\ 0.5, & \text{for } j = J \end{cases} \tag{5.10a}$$

$$c_n = 2b_j. \tag{5.10b}$$

5.2 Implementation of the Local Stability Solver

In the following section the solution of the local stability problem will be presented. Therefore, a frequency f or ω and a spanwise wavenumber β are given as real quantities and the spacial problem will be solved to obtain the complex streamwise wavenumber α.

5.2.1 The Eigenvalue Solver

The Chebyshev polynomials are used to solve the eigenvalue problem stated by the Orr-Sommerfeld and Squire equation (2.136) and (2.137). By solving the eigenvalue problem for a given frequency f and spanwise wavenumber β the eigenvalues and eigenvectors are obtained. The eigenvalues are plotted as real and imaginary values as shown in Figure 5.2. The Tollmien-Schlichting eigenvalue has to be selected from this spectrum. By performing the eigenvalue calculation for two different number of collocation points numerical solutions can be compared and values that are different can be excluded. Note that all values slightly differ for the two calculations and a tolerance has to be given. That tolerance is used to

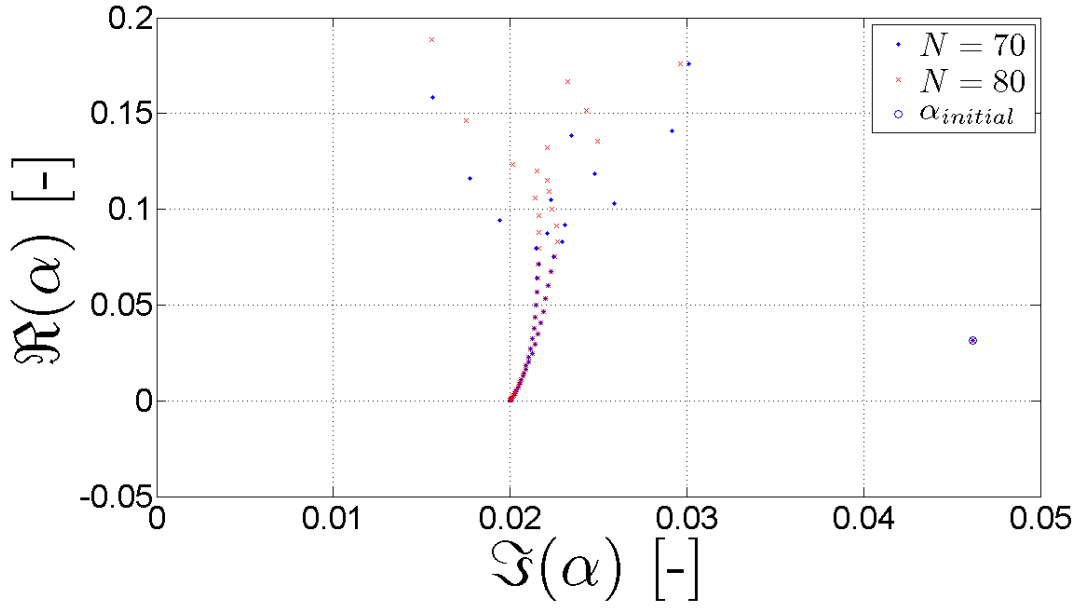


Figure 5.2.: The eigenvalues α are plotted for two different numbers of collocation points to find an initial guess for the Tollmien-Schlichting mode.

exclude values that have a large difference. Further, eigenvalues that contain a negative real value are excluded as they are not physical. Thereby, a large amount of "unphysical" eigenvalues is excluded and only a few are left to pick the Tollmien-Schlichting wavenumber.
The numerical implementation to solve for all eigenvalues is explained in detail in the work of Reeh [17] and [18]. That solver is also used here to find the initial guess. In Figure 5.2 the Tollmien-Schlichting eigenvalue is found on the right side.
After having obtained a good guess for the complex streamwise wavenumber α the iteration process can be started. First of all, the quantities obtained for the mean flow have to be rescaled to fit on the grid used for the stability analysis. The rescaling is achieved by interpolation and extrapolation of the mean flow quantities. The quantities for the mean flow are computed by using a solver described in section 3.4.1 or 3.4.2. Note that Chebyshev discretization is employed and that the first entry of the Chebyshev matrix belongs to the grid point in the freestream. Therefore, the velocity profiles need to be flipped to satisfy that precondition. The first and second order Chebycheff differentiation matrices are only applied to the perturbation quantities that occur in (2.139).
In combination with the quantities from the mean flow that are adjusted to the stability solver grid, the matrix $\boldsymbol{L}$ given in equation (2.145) with the submatrices given in (2.146) can be computed. An efficient way to set up the $4J$x$4J$ matrix in *MATLAB* is given by the command `diag` to convert a vector into a diagonal matrix or the multiplication with the identity matrix which is created by the command `eye`. Thereby, each submatrix is conveniently computed resulting in the eigenvalue problem given in symbolic structure

$$\begin{pmatrix} \boldsymbol{L}_{1,1} & \boldsymbol{L}_{1,2} & \boldsymbol{L}_{1,3} & \boldsymbol{L}_{1,4} \\ \boldsymbol{L}_{2,1} & \boldsymbol{L}_{2,2} & \boldsymbol{L}_{2,3} & \boldsymbol{L}_{2,4} \\ \boldsymbol{L}_{3,1} & \boldsymbol{L}_{3,2} & \boldsymbol{L}_{3,3} & \boldsymbol{L}_{3,4} \\ \boldsymbol{L}_{4,1} & \boldsymbol{L}_{4,2} & \boldsymbol{L}_{4,3} & \boldsymbol{L}_{4,4} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{u}} \\ \hat{\boldsymbol{v}} \\ \hat{\boldsymbol{w}} \\ \hat{\boldsymbol{p}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{5.11}$$

Finally, the boundary conditions (2.134) need to be implemented into the matrix $\boldsymbol{L}$. The first and last line of the submatrices $\boldsymbol{L}_{2,1}$ and $\boldsymbol{L}_{3,2}$ are set to zero and the first position in the first line and the last position in the last line is set to one. That ensures the $\hat{u}$ and $\hat{v}$ at the wall and in the free stream are zero. In $\boldsymbol{L}_{4,3}$ the first line is set to zero and then the first position is set to one ensuring that $\hat{w}$ in the free stream becomes zero. The last boundary condition, $\hat{w}$ at the wall, is the iteration condition used in the following. However, as only five boundary conditions are set so far and a sixth condition is needed, a normalization condition is introduced. The derivative $\partial\hat{u}/\partial y$ at the wall is set to one and thus the last row of $\boldsymbol{L}_{4,1}$ is replaced with the last row of the Chebyshev matrix for the first order differential. In order to ensure that the derivative becomes one, the right-hand side of that row in $\boldsymbol{L}$ has to be set to one.
Finally, the system shown in (5.11) is solved in MATLAB resulting in the eigenvectors for the inserted α. The value of $\hat{w}$ at the wall will be used as a iteration condition in the following secant method. The secant method is a Newton-like method that approximates the derivative using a secant. In that way the exact derivative does not have to be calculated which reduces the computational costs. The loss in accuracy is compensated by introducing an older guess to compute the derivative. The iteration condition reads

$$\alpha^{(i+1)} = \alpha^{(i)} - \frac{\alpha^{(i)} - \alpha^{(i-1)}}{\hat{w}_{(0)}^{(i)} - \hat{w}_{(0)}^{(i-1)}} \hat{w}_{(0)}^{(i)}. \tag{5.12}$$

The iteration is stopped if $\hat{w}_{(0)}^{(i)} < 10^{-10}$ and the actual α is returned, which is then used as an initial guess for the next position.

5.2.2 Computation of Growth Rate and eN-Method

After having obtained the complex eigenvalues for each streamwise position the growth rate can be plotted. The growth rate is defined as

$$\sigma^* = \Im(-\alpha^*) \tag{5.13}$$

and for positive values of σ^* the Tollmien-Schlichting modes will grow and finally initiate transition. For negative values of σ^* the perturbations are damped and the neutral stability point is found for $\sigma^* = 0$. In Figure 5.3 the growth rate for a Blasius case is exemplary shown. Typically, the Tollmien-Schlichting waves are significantly damped at the beginning and the damping decreases with increasing streamwise position. The similarity flows are characterized by a range of growing instability which is then followed again by a stable range. The dimensionless frequency which is usually used to plot similarity flows is defined as

$$F = \frac{2\pi f^* \nu^*}{{U_{ref}^*}^2} 10^6 = \frac{2\pi f^*}{Re} \frac{L_{ref}^*}{U_{ref}^*} 10^6 \quad \Leftrightarrow \quad f^* = \frac{ReF}{2\pi} \frac{U_{ref}^*}{L_{ref}^*} 10^{-6} \tag{5.14}$$

where f^* is the frequency in Hz, L_{ref}^* and U_{ref}^* are the local reference scales for the length and the velocity. Further, Re is the local Reynolds number based on the reference values. In the following calculations the Blasius length scale and the local edge velocity are used as reference scales. The advantage of using the dimensionless frequency F is that different calculations can be compared with each other.
The growth rate can then be utilized to find the neutral stability curve. That curve gives frequencies and the corresponding streamwise positions that are fanned. Those curves are characterized by a minimum streamwise position from where the first frequencies fan the flow. Before that minimum streamwise position no growth can be found. In Figure 5.4 neutral curves are plotted for different Hartree parameters.

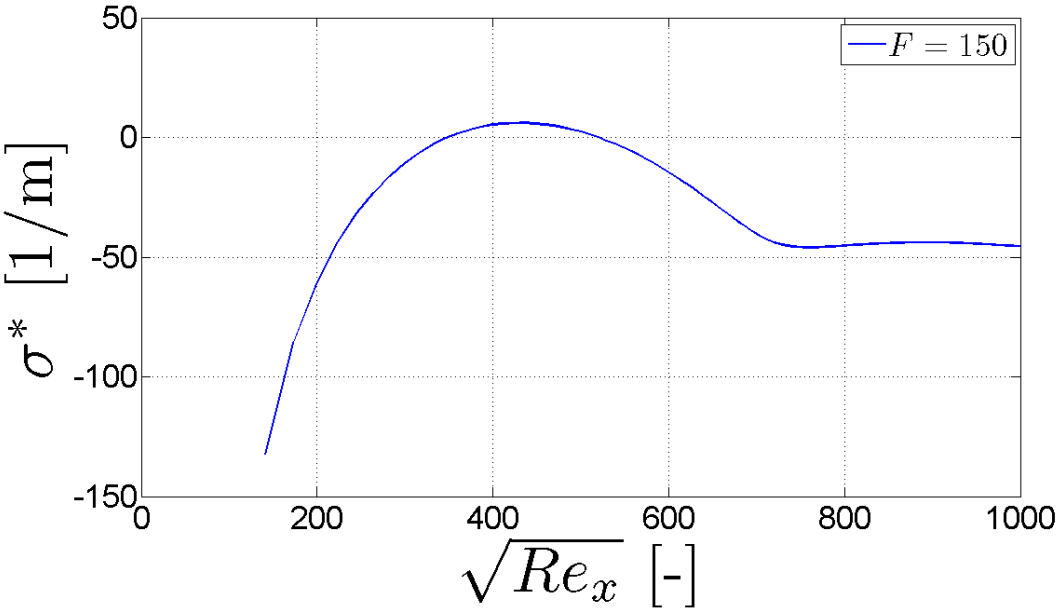


Figure 5.3.: Growth rate for a Blasius flow with $F = 150$.

For negative values the flow decelerates and becomes less stable. Therefore, an explanation is found why a small wall-normal derivative of the streamwise component of the mean flow results in less stable flows. In contrast, for increasing β_H the flow is accelerated and the neutral stability curves are shifted to higher Reynolds numbers and lower frequencies. The profiles for those Hartree parameters become fuller and thus are more stable.

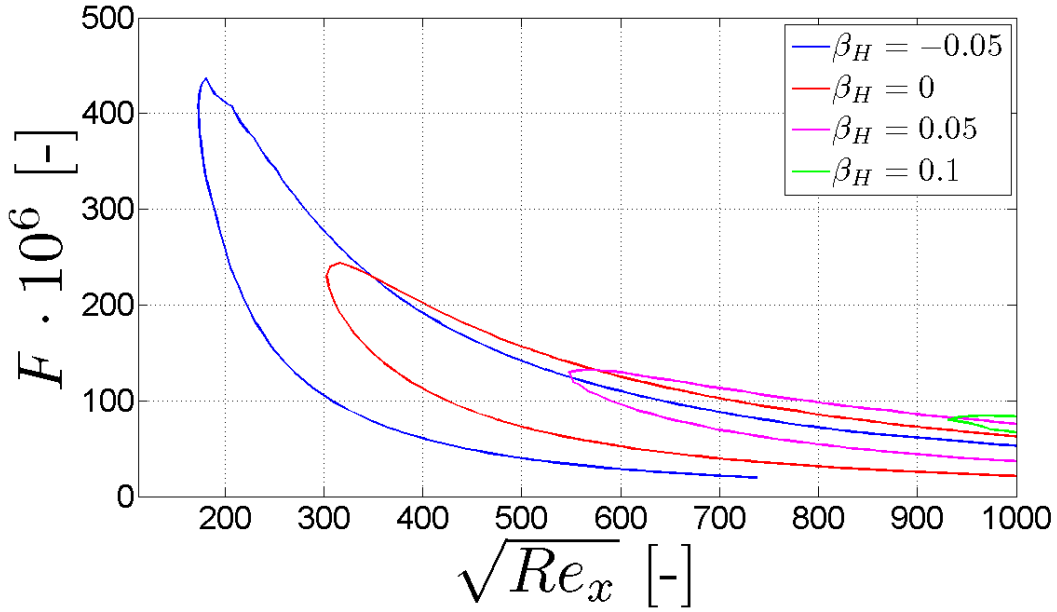


Figure 5.4.: Neutral stability curves for different Hartree parameters.

In order to determine a transition point that matches with the onset of turbulence an empirical method has been developed. The method was developed by van Ingen in 1956 [24]. The N-factor is defined as

$$N = \ln\left(\frac{A}{A_0}\right) = \int_{x_0}^{x} -\Im(\alpha)\, dx \tag{5.15}$$

where A is the amplification factor corresponding to an initial wave amplitude A_0 that refers to the start of the stability analysis. The integral of the growth rate is calculated and transition is assumed to take place when a certain prescribed N-factor is reached. The prescribed value for N is empirical since the method does not account for the receptivity process and the nonlinear growth of disturbances. The N-factor is logarithmic and refers to e^N. The common value often used in literature for the onset of transition is $N_{trans} = 9$ which then results to an amplification of $e^9 \approx 8\,130$ compared to the initial value. In Figure 5.5 and 5.6 the N-factors for the pressure and suction side are plotted for the same flow paramters as in section 3.4 and for different frequencies. The mean flow is calculated by the direct solver shown in section 3.4.1. Note that the amplification is much larger for the suction side and thus the value of $N = 9$ (compare with the black line in Figure 5.6) is reached already at $x_c = 0.4$. In contrast, the critical amplification is not reached for the pressure side. The streamwise position of the last calculation refers to the point where the boundary-layer code does not converge anymore. The boundary-layer method used to generate the mean flow data is the direct method because the agreement with the reference data of KTH is best. However, as already mentioned before different reference data sets should be used to compare.

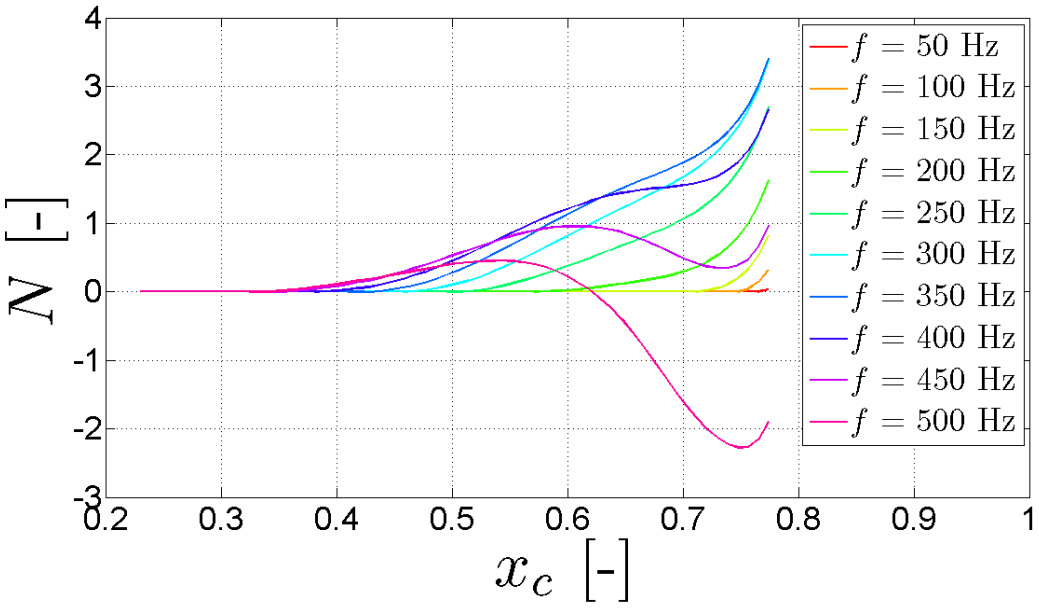


Figure 5.5.: N-factor for the pressure side (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

The maximum N-factor at each streamwise position can be used to define the critical envelop that characterizes maximum growth at each position. The calculation is performed for some arbitrarily chosen frequencies. As a next step the measurement data from free-flight experiments can be used to investigate the growth of the Tollmien-Schlichting waves for frequencies measured under experimental conditions.

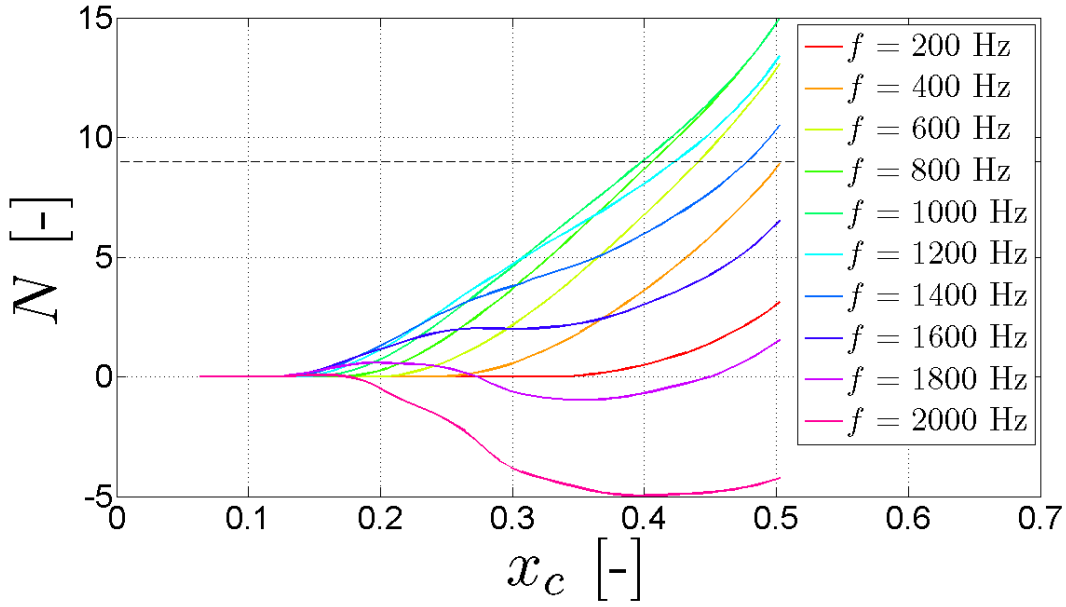


Figure 5.6.: N-factor for the suction side (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

Frequencies that lead to maximum amplification are found in the range of 300 to 400 Hz for the pressure side. For the suction side the frequencies are much higher as maximum amplification is found for 800 to 1200 Hz.

5.2.3 Comparison with an Empirical Envelope Method

For the comparison of the obtained N-factor curves, an empirical way to determine the envelope of the N-factor is given by Drela and Giles [11]. The great advantage of prediction the envelope by an empirical formula is the saving in computational time. The computation of the N-factor curves needs some experience as the unstable frequencies are not known a priori. Further the calculation has to be repeated for each frequency which can be very time consuming and thus an empirical method could save a lot of time.
First of all, the empirical method is tested for Falkner-Skan profiles. The envelope of the N-factor is approximated by

$$N = \frac{dN}{dRe_{\delta_2}}(H_{12}) \left[Re_{\delta_2} - Re_{\delta_2,0}(H_{12}) \right] \tag{5.16}$$

where Re_{δ_2} is the Reynolds number based on the momentum thickness and $Re_{\delta_2,0}$ is the critical Reynolds number where first frequencies become unstable. The empirical method is based on the shape factor H_{12} on which all quantities depend. The envelope for Falkner-Skan profiles is approximated by linear lines with the slope $\frac{dN}{dRe_{\delta_2}}$ which was empirically determined to be

$$\frac{dN}{dRe_{\delta_2}} = 0.01 \sqrt{\left[2.4 H_{12} - 3.7 + 2.5 \tanh\left(1.5 H_{12} - 4.65 \right) \right]^2 + 0.25}. \tag{5.17}$$

The critical Reynolds number is found empirically to be

$$\log_{10} Re_{\delta_2,0} = \left(\frac{1.415}{H_{12} - 1} - 0.489 \right) \tanh\left(\frac{20}{H_{12} - 1} - 12.9 \right) + \frac{3.295}{H_{12} - 1} + 0.44. \tag{5.18}$$

In *MATLAB* the critically Reynolds number is found by applying the nonlinear root finder `fzero`.
The results for the Falkner-Skan profiles is depicted in Figure 5.7. The empirical approximation does not perfectly match with the computed N-factors. For increasing Hartree paramters the approximation becomes worse and the good agreement as shown in [11] cannot be affirmed. A possible explanation could be the computation of the shape factor which is very sensitive to small differences of the displacement or momentum thickness. The calculation of the data used is based on numerical integration using the a trapezoidal method. Thereby, small differences compared to the values given in literature [20] are observed. The reason for the employment of the numerical integration is that the results can be compared to other boundary-layer calculations directly. In contrast, if the the calculation of the boundary-layer characteristics is accomplished by using equations (2.89) and (2.90) the quantities match perfectly with literature.

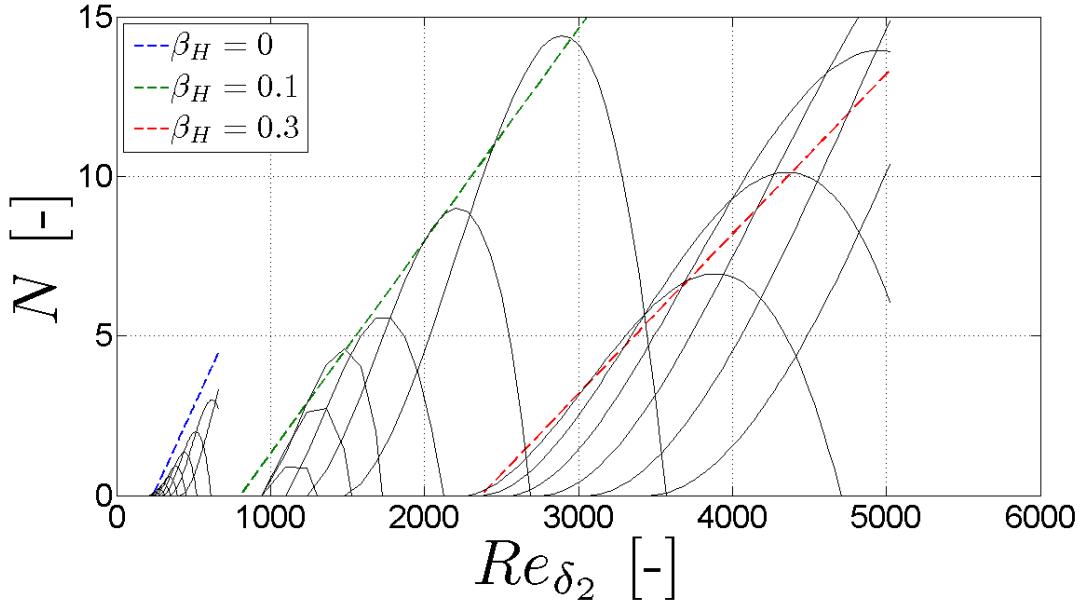


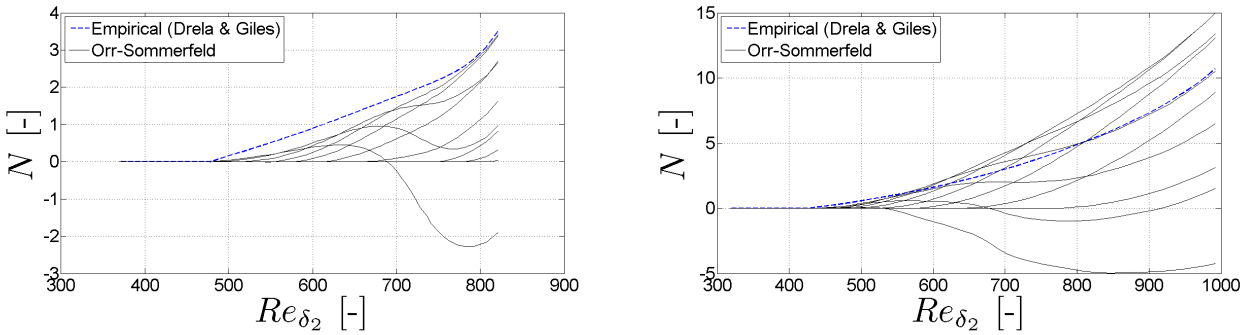
Figure 5.7.: N-factor for different Hartree parameters using the formula of Drela and Giles [11].

The empirical method can also be applied to airfoil flows. In that case the slope changes for each position as the shape factor varies along the chord. The N-factor is then found by integration of the slopes along the chord such that

$$N = \int_{Re_{\delta_2,0}}^{Re_{\delta_2}} \frac{dN}{dRe_{\delta_2}} dRe_{\delta_2}. \tag{5.19}$$

The critical Reynolds number has to be set manually.
In Figure 5.8 the results of the empirical method are plotted in comparison with the results obtained by the stability analysis described above. The critical Reynolds number is chosen in the way that it matches with the first Reynolds number of the Orr-Sommerfeld solver that becomes unstable. The results for the pressure side is shown in Figure 5.8a and it proves that the empirical method is able to predict the envelope. The rise in the slope at the end is very well detected.
In Figure 5.8b the results for the suction side are shown. Again the empirical method is able to detect the envelope even though the least unstable frequencies of the Orr-Sommerfeld shown a larger amplification. However, the method is only empirical and thus some investigated frequencies might not be included

in the empirics. Due to that the prediction for the suction side is not as good as for the pressure side. Further, the amplification rates for the suction side are much larger than for the pressure side and the empirical prediction might become worse for increased amplification factors.



(a) Pressure side

(b) Suction side

Figure 5.8.: Comparison of N-factors between empirical method of Drela and Giles [11] and explicit numerical computation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

5.3 Implementation of a Solver for the Parabolized Stability Equations

The implementation of the stability equations is very similar to the linear stability analysis above. The calculation is started with an initial streamwise wavenumber α that is obtained from the eigenvalue finder. Note that the derivative in the streamwise direction of the perturbation pressure is not included as the term causes numerical problems because the equations then become elliptic.
After having obtained an initial guess the local Orr-Sommerfeld solver is used to iterate the complex streamwise wavenumber at the first position. The iteration process is described above in section 5.2. The kinetic energy of the disturbances are normalized to one such that

$$\hat{\boldsymbol{q}}_{\text{init}} = \frac{\hat{\boldsymbol{q}}}{\left\|\hat{\boldsymbol{q}}\right\|_E} \tag{5.20}$$

where the energy norm $\left\|\hat{\boldsymbol{q}}\right\|_E$ is given by

$$\left\|\hat{\boldsymbol{q}}\right\|_E = \left(\frac{1}{2}\hat{\boldsymbol{q}}^T \boldsymbol{W} \hat{\boldsymbol{q}}\right)^{1/2}. \tag{5.21}$$

The integration is replaced by a matrix multiplication of the Chebyshev integral weight matrix given in equation (5.9). Note that the perturbation pressure is not included in the norm. The quantity $\hat{\boldsymbol{q}}$ is a large vector the has the size of four times the number of collocation points.
The initial eigenvector $\hat{\boldsymbol{q}}_{\text{init}}$ is then used as an input for the iteration process along the chord. For the second streamwise position the matrix $\boldsymbol{L}$ from equation (2.145) is build. The PSE includes also the streamwise derivative and thus the eigenvectors of former positions have to be included as well. The equation that has to be solved thus becomes

$$\boldsymbol{L}^{(i)}\hat{\boldsymbol{q}}^{(n),(i)} = -D_{x,n-1}\boldsymbol{D}\hat{\boldsymbol{q}}^{(n-1)} - D_{x,n-2}\boldsymbol{D}\hat{\boldsymbol{q}}^{(n-2)}. \tag{5.22}$$

The matrix $\boldsymbol{L}$ included the backward differential coefficient $D_{x,n}$ in front of the matrix $\boldsymbol{D}$. The backward differential coefficients are set up by equation 3.22 as described in section 3.1.3. The iteration count is marked by the superscript i. The system is then solved for $\hat{\boldsymbol{q}}^{(n),(i)}$. In contrast to the boundary-layer calculation the differential operators are build by using Chebyshev discretization and thus the differential matrices are not sparse but fully populated.
In the next step the auxiliary function given by equation (2.148) is used to ensure that the fast oscillations are captured by the exponential part and the slow variations by the shape function. Again the pressure term is not included in the integration. The integral is computed by a matrix multiplication such that

$$f_{\text{aux}} = \int_0^\infty \hat{\boldsymbol{q}}^H \frac{\partial}{\partial x}\hat{\boldsymbol{q}}\, dy = \hat{\boldsymbol{q}}^{(n),(i)^T} \boldsymbol{W} \frac{\partial \hat{\boldsymbol{q}}^{(n),(i)}}{\partial x} \quad \text{with} \quad \frac{\partial \hat{\boldsymbol{q}}^{(n),(i)}}{\partial x} = D_{x,n}\hat{\boldsymbol{q}}^{(n),(i)} + D_{x,n-1}\hat{\boldsymbol{q}}^{(n-1)} + D_{x,n-2}\hat{\boldsymbol{q}}^{(n-2)}. \tag{5.23}$$

The quantity is normalized with the maximum absolute value of $\boldsymbol{u}^{(n),(i)}$ of the shape function squared.
The new iterate for the streamwise wavenumber α is then computed by the secant method. It reads

$$\alpha^{(i+1)} = \alpha^{(i)} - \frac{\alpha^{(i)} - \alpha^{(i-1)}}{f_{\text{aux}}^{(i)} - f_{\text{aux}}^{(i-1)}} f_{\text{aux}}^{(i)}. \tag{5.24}$$

The solution is supposed to be converged if the absolute value of $f_{\text{aux}}^{(i)}$ is less the 10^{-8}.
After convergence the iteration process starts for the next position again till the last boundary-layer profile.

5.3.1 Computation of Growth Rate and eN-Method

In contrast to the Orr-Sommerfeld solver the growth rate is based on the norm of the disturbance energy which is given in equation (5.21). The growth rate is then given as

$$\sigma_E^{(n)} = -\Im(\alpha^{(n)}) + \Re\left(\frac{\ln\left(\left\|\hat{\boldsymbol{q}}^{(n)}\right\|_E\right) - \ln\left(\left\|\hat{\boldsymbol{q}}^{(n-1)}\right\|_E\right)}{x^{(n)} - x^{(n-1)}}\right). \tag{5.25}$$

but only the real parts are assumed to characterize the growth rate. The dimensional value is found be dividing by the length scale.
In the following the effect of taking into account the changes in streamwise direction are shown for the Blasius case. The linear stability analysis from section 5.2 is compared with the results obtained by the PSE as described above. In Figure 5.9 the neutral stability curve is plotted for different dimensionless frequencies F over the square root of the Reynolds number. It is clearly visible that the unstable region is larger when the non-parallel effects are taken into account. The neutral point where first frequencies lead to instabilities is moved from 300 to about 280 for the square root of the Reynolds number. That means the non-parallel effects destabilize the flow. The wavy results for high Reynolds numbers and low frequencies as shown in Figure 5.9 are caused by numerical instabilities. The lowest frequency investigated is $F = 20$ and thus the discontinuities occur. For higher Reynolds numbers the differences between Orr-Sommerfeld and PSE decrease because non-parallel effects are diminished.
Corresponding to the definition of the growth rate the N-factor is also based on the energy and it becomes

$$N_E = \int_{x_0}^{x} \left(\sigma_E^{(n)}\right) dx. \tag{5.26}$$

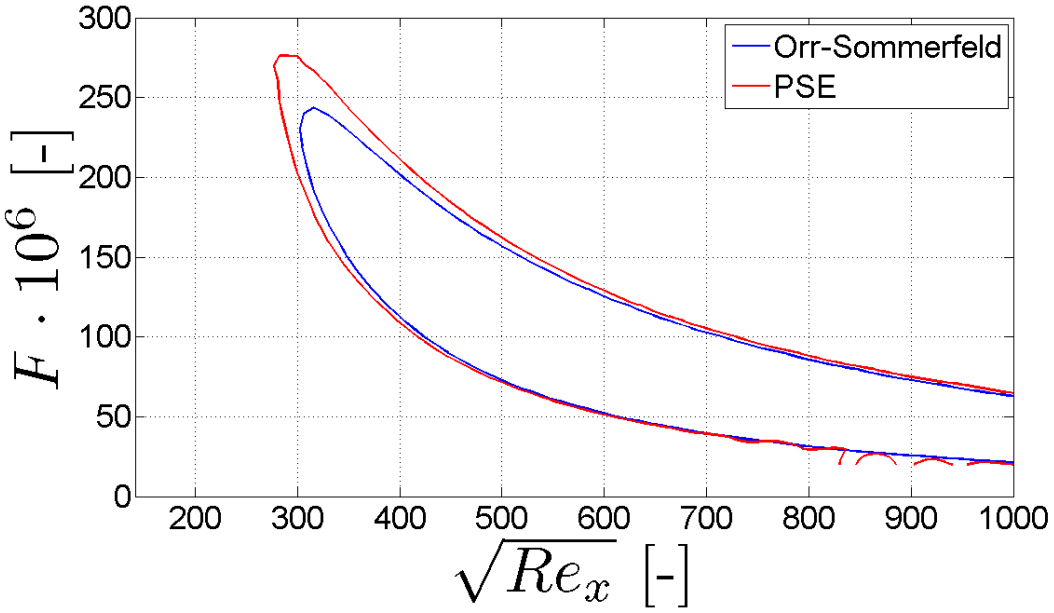


Figure 5.9.: Comparison between Orr-Sommerfeld and PSE solver for the neutral stability curve of a Blasius flow.

Finally, the N-factors for the *MW-airfoil* are shown. The boundary-layer calculations for the mean flow are based on the Reynolds number of $Re = 2\,351\,842$ and an angle of attack of $\alpha = 3.3°$. In Figure 5.10 the N-factors for the pressure side are shown. The results are very similar to the local stability calculations presented in Figure 5.5. Like already observed for the neutral stability curve of the Blasius flow the instabilities are slightly higher. The maximum value for the N-factor is about 3.8 for the pressure side and also the envelope describing maximum growth at each positions is observed a little higher than for the local theory. The frequencies that lead to the greatest instabilities are in the range of 300 to 400 Hz.

Similar results are also obtained for the suction side as shown in Figure 5.11. The frequencies leading the maximum amplification lay in the range between 800 and 1200 Hz. The critical N-factor of 9 is crossed for a streamwise position of about $x_c = 0.38$ which is slightly before the results obtained by local stability analysis.

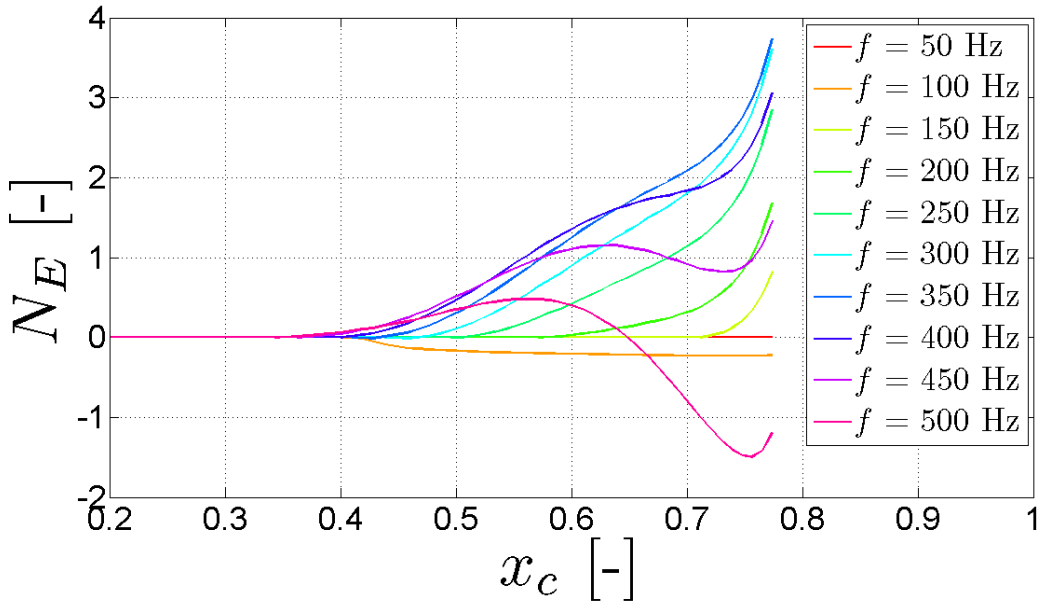


Figure 5.10.: N-factor for the pressure side (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

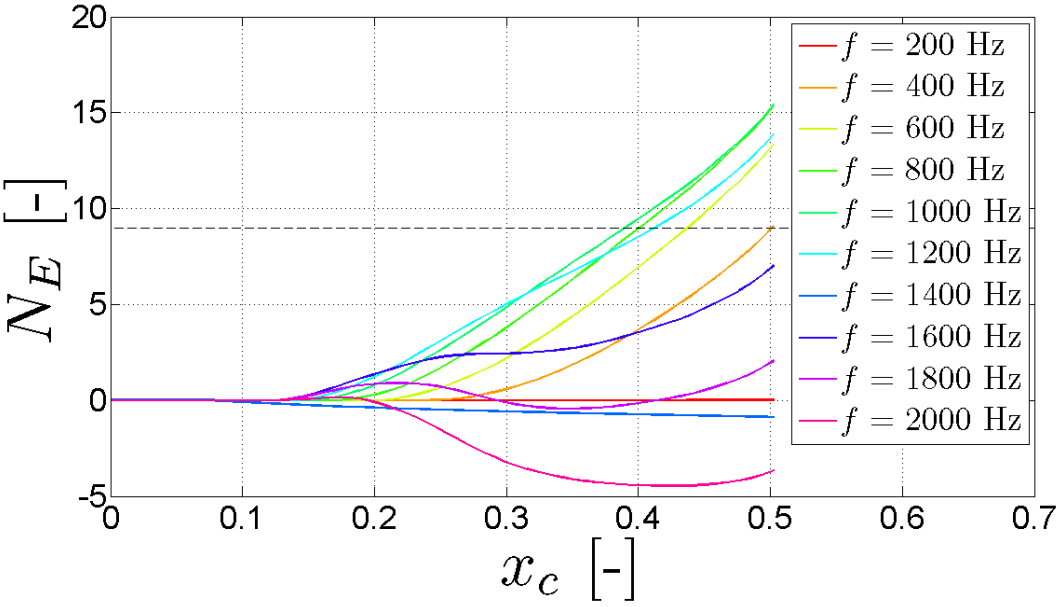


Figure 5.11.: N-factor for the suction side (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$).

6 Conclusions and Outlook

In the present work different methods were presented to solve the boundary-layer equation. As a first approximation the integral boundary-layer equation based on the Pohlhausen method was implemented. The Walz method was implemented and fairly good results were obtained. Similar results were found for the implementation of Thwaites' method. The boundary-layer profiles in such methods could be found which are based on a polynomial approximation (the Pohlhausen polynomial is quartic). More elaborate methods to solve the boundary-layer equations are necessary. In general, those integral boundary-layer methods can only find quantities for the boundary-layer characteristics and due to little computational time they are still present to solve for turbulent flows. This flow field is not sufficient for stability analysis and advanced transition prediction.

In order to find the exact values of the boundary-layer profiles the direct boundary-layer equations have to be solved. The equation is a partial differential equation which complicates a direct solution. The introduction of boundary-layer transformations simplifies the solution. The Falkner-Skan transformation is very common to change the spacing in wall-normal direction. Thereby, the boundary-layer equation can be solved directly in a very efficient way. In general, the boundary-layer equation for the laminar part of an airfoil is done in less than ten seconds and the result matches almost perfectly with the reference data obtained by a compressible solver from KTH Stockholm. In comparison to the integral boundary-layer solver the boundary-layer profiles were used to compute the boundary-layer characteristics. In the case of the direct solution of the boundary-layer equation the characteristics were obtained by numerical integration. The agreement with the reference data computed by *XFOIL* is very close. Small discrepancies occur due to the fact that *XFOIL* solves the boundary layer interactively and thus updates the pressure distribution after each iteration. Due to that small discrepancies between the boundary-layer calculation occur as the converged solution of *XFOIL* is based on a slightly different pressure distribution. In addition, XFOIL uses an integral method that is less accurate.

In contrast to the direct solution of the boundary-layer equation the boundary-layer profiles can also be obtained by introducing a stream function and then solve for the dimensionless stream function $f(x, \eta)$. In that case a third order differential equation was derived that could be reduced to a system of three first order differential equations. In the present work the derivation of such a system was shown for the Falkner-Skan transformation and the Görtler transformation. The latter has the special feature that the wall-normal dimensionless coordinate takes the boundary-layer growth into account which is favorable for numerics. Both transformations were implemented into a numerical solver that is based on the Keller box discretization. Instead of building the derivatives around one grid point the derivatives are centered around four grid points. Thereby, the derivatives become smoother and in general the numerics are more stable.

The great advantage of the solution of the direct boundary-layer equation is the finding of the velocity profiles which can be used to investigate the stability of the flow and thus find the transition point. However, as the solvers are derived for the laminar part only, the computation stops at the separation point when the adverse pressure gradient becomes too large. That point is characterized by the Hartree parameter $\beta_H = -0.19884$. Especially if the boundary-layer solver should be coupled with some other solver and the computation of the whole airfoil is needed, the divergence of the solver is not desirable. Therefore, the Cebeci and Smith zero-equation turbulence model was introduced that computes the tur-

bulent part of the flow over the airfoil. Note that the turbulence model does not solve any additional transport equation but is based on empirical functions for the inner and outer region of the turbulent boundary layer. In the present work the turbulence model was compared to the results of *XFOIL* and differences are noticeable. However, the *XFOIL* calculation is also based on an empirical turbulence model and thus does not hold as qualitatively good reference. The computed boundary-layer profiles have not been evaluated yet due to lack of reference data. In the future the boundary-layer calculation results for the mean flow have to be compared to some reference data obtained by more elaborate RANS or LES/DNS computations to check if the boundary-layer profiles are correct.
In contrast to the solution of the boundary-layer equation that is based on the pressure distribution from an inviscid solver (referred to as the *standard method*) a more elaborate way of computing the boundary layer has been investigated. In the so-called *inverse method* the displacement thickness was used to update the edge velocity (that can be directly computed out of the pressure distribution). Thereby, the singularity where the direct solvers diverge is avoided and the computation can be performed for the whole airfoil. The quasi-simultaneous method by Veldman [25] solves for the displacement thickness and the edge velocity simultaneously in one iteration step. Thereby, very fast convergence can be obtained and the number of iterations in a viscous-inviscid interaction method can be reduced significantly. The implementation is based on a Fortran code by Cebeci [6] that is converted into *MATLAB*. The reason for the implementation of the external code is the additional solver for the wake which will be needed in the future to compare the numerical results to experimental data obtained in free-flight experiments. However, not every part of the code is fully understood yet and only first validation calculations have been performed. The results for the displacement and momentum thickness match perfectly with the *XFOIL* reference for the laminar part. The velocity profiles are not compared to any reference data yet. Nevertheless, the main features of Veldman's method are found in the code and the turbulence model is also based on Cebeci and Smith.
As the codes written for turbulent flows need the information where the turbulence model should be used, a transition points needs to be set. In the calculations presented in the work the transition point was set manually to match with the reference data from *XFOIL*. In order to reduce that "arbitrariness" the prediction of the transition point should be included by stability analysis. For two-dimensional flows Tollmien-Schlichting waves are the least stable modes that first lead to transition. In order to find an unstable frequency that fan the Tollmien-Schlichting modes the eigenvalues of a stability analysis were investigated. In stability analysis the flow computed by the boundary-layer solver is used as a mean flow and in the present work small disturbances are superimposed. The influence of those disturbances was investigated and disturbance frequencies destabilizing the flow could be found.
In a first step the local stability equations were solved to find the spatial amplification that leads to transition. The local stability equations are formed by the Orr-Sommerfeld and Squire equations. Those equations do not take into account the wall-normal velocity and the derivative of the velocities in streamwise direction. In other words, the flow is assumed to be parallel which of course is not the case in a real flow as the boundary-layer thickness increases in streamwise direction. The eigenvalues of those equations were numerically computed by a method shown in [17]. In order to find the eigenvalue that belongs to a Tollmien-Schlichting wave a filter is implemented that finds the correct eigenvalue. That filter has to be manually operated till the correct value is found. In the following the eigenvalue is used as an initial guess for a Newton-like method to compute the progress of the eigenvalue. The advantage of that iteration is a significantly reduced computation time as only one eigenvalue is solved for. Finally, the growth rate of that eigenvalue is calculated and the e^N-method of van Ingen [24] is applied to find the amplification rate. This method can be used to find the transition position that is empirically found for an N-factor between 8 and 10. Further, those results are compared to an empirical method of Drela

and Giles [11] to predict the transition point and agreement is fairly well.
Complementary to the local stability equations are replaced by the so-called parabolized stability equations that also take into account the non-parallel effects. The results of such calculations thus represent the physics of a real flow in a closer agreement to reality. However, those equations introduce a slightly elliptic behavior into the stability equations which results in unstable numerics. In order to relax the equations the streamwise derivative of the pressure perturbation is omitted. The computed neutral stability curve for a Blasius flow matches quite well with results shown in [21] and thus the parabolized stability equations are applied to the investigated airfoil flow. The numerics were less stable than for the local stability equations and the results for the airfoil flow only slightly differed from the results of the local analysis. However, the results of the stability analysis have not been compared with reference data yet and that definitely should be carried out in the future.
After having solved all difficulties with steady boundary-layer calculation and transition prediction a similar method has to be found that can be used for unsteady airfoil flows. In [7] an extension of the standard and indirect method is shown that can be applied for unsteady flows. The discretization is again based on the Keller box and a Falkner-Skan transformation is used to transform the wall-normal direction. In order to solve for the unsteady problem a third dimension has to be introduced which states the time axis. The Keller box which is a square in the 2D case becomes a cube for the unsteady flow which is used for centering. The additional third dimension adds some numerical problems as mentioned in [7]. Probably the first obstacle to overcome will be the definition of numerically stable initial conditions for the (x,y) and the (t,y) planes to avoid numerical difficulties. As the stagnation point varies with time the initial velocity profile will contain the derivative $\frac{\partial u}{\partial t}$ which has to be determined. Further, the derivative of the edge velocity will also change with time which leads to a more complex system. More information to the implementation of an unsteady boundary solver can be found in [7].
A first step to implement an unsteady method to predict the flow for the *MW-airfoil* in the case of changing angles of attack might be the use of the unsteady Hess-Smith-Panel method that is already available at the institute. Thereby, the inviscid flow will be solved unsteady and the steady boundary-layer method derived in the shown work can be used to solve the boundary-layer equations and to predict the transition point. The unsteady effect will then be included in the edge velocity distribution. Comparisons with experimental data will then show if further alteration of the boundary-layer code are necessary.

A Input File Format for Boundary-Layer Calculation

The input format for the boundary-layer calculation is shown in the following. The input format can also be used for the compressible KTH boundary-layer code. The first row is used as a global input. The reference length `Lref` is used as a scaling for the airfoil coordinates. The combination of the Mach number `Mach` and the temperature at infinity `T_inf` is used to calculate the free-stream velocity U_∞

$$U_\infty^* = Ma_\infty c_\infty^* = Ma_\infty \sqrt{\gamma R^* T_\infty^*} \tag{A.1}$$

where γ is the ratio of specific heats and R is the specific gas constant which are both assumed to be known constants. The free-stream velocity is then used in combination with the reference length and the Reynolds number `Re` to obtain the kinematic viscosity ν. The sweep angle of the airfoil `Sweep_angle` is set to zero as only a two-dimensional flow is investigated.
The second set of data is given below the global input. The data contains the airfoil geometry which is given by `x/Lref` and `z/Lref`. Note that the coordinate z is the wall-normal coordinate y. The KTH code uses z as wall-normal coordinate and y as spanwise coordinate. Further, the pressure distribution `cp` has to be given for each position. The data set of `cq` belongs to the suction rate at the wall which is zero for the case investigated in the work here and the dimensionless wall-temperature `Tw` is also zero which resembles the adiabatic wall condition.
In the following an example for the input data formatting is shown. The values are separated by a tab. Note that the list is not completely shown here and more grid points would follow below.

```
#Lref Re Mach T_inf Sweep_angle
   1.3500000000e+000   2.3518428079e+006   9.7702778945e-002   2.7578144163e+002   0.0000000000e+000
#x/Lref z/Lref cp cq Tw
   3.9617351724e-003  -8.0944673868e-003   1.0000000000e+000   0.0000000000e+000   0.0000000000e+000
   4.7570000000e-003  -8.7090000000e-003   9.9677000000e-001   0.0000000000e+000   0.0000000000e+000
   5.8960000000e-003  -9.5010000000e-003   9.8313000000e-001   0.0000000000e+000   0.0000000000e+000
   7.1380000000e-003  -1.0274000000e-002   9.6338000000e-001   0.0000000000e+000   0.0000000000e+000
   8.4840000000e-003  -1.1035000000e-002   9.3828000000e-001   0.0000000000e+000   0.0000000000e+000
   9.9420000000e-003  -1.1790000000e-002   9.1208000000e-001   0.0000000000e+000   0.0000000000e+000
   1.1520000000e-002  -1.2543000000e-002   8.8365000000e-001   0.0000000000e+000   0.0000000000e+000
   1.3230000000e-002  -1.3299000000e-002   8.5469000000e-001   0.0000000000e+000   0.0000000000e+000
   1.5087000000e-002  -1.4064000000e-002   8.2614000000e-001   0.0000000000e+000   0.0000000000e+000
   1.7107000000e-002  -1.4842000000e-002   7.9622000000e-001   0.0000000000e+000   0.0000000000e+000
   1.9313000000e-002  -1.5636000000e-002   7.6590000000e-001   0.0000000000e+000   0.0000000000e+000
```

For airfoil flows the first grid position belongs to the stagnation point. The routine that creates such formatted files out of a given pressure distribution and airfoil geometry is given in the folder “Create_BL_Test_Files”. The routine computes input files for the pressure and suction side calculation. Further, it distinguishes between an incompressible and compressible solver. As the KTH solver is compressible the pressure distribution at the stagnation point has to be amended in order to excluded the possibility of cross flow. The C_p-value at the stagnation point is then calculated by

$$C_p = \left[\left(1 + \frac{\gamma+1}{2} Ma_\infty^2 \right)^{\frac{\gamma}{\gamma-1}} - 1 \right] \frac{2}{\gamma Ma_\infty^2} \tag{A.2}$$

The boundary-layer codes are written for an input format that is given above.

B Output Format of the Boundary-Layer Codes

Likewise the input format for the boundary-layer calculation which is shown in Appendix A, a suitable input format for the transition prediction calculations has to be defined. Each boundary-layer code uses the same formatting. The profiles and their derivatives for each position are saved as a cell array in *MATLAB* and the quantities are scaled with a length and a velocity scale. Note that the profiles are saved in the same format as in the KTH solver and the y and v are the quantities for the spanwise direction and z and w are the values for the wall-normal direction.
The Blasius length scale

$$y^*_{\text{Blasius}} = L^*_{ref}\sqrt{\frac{x}{ReU_e}} \tag{B.1}$$

is used as the length scale and the edge velocity U^*_e is used as the velocity scale. The Reynolds number that is used for the transition prediction is also defined locally with those two values as

$$Re = \frac{y^*_{\text{Blasius}}U^*_e}{\nu^*}. \tag{B.2}$$

The scaled mean flow is then saved in the cell array `mean` and the reference values are saved in the cell array `ref`.
Further, the boundary-layer characteristics are used as an output as well as the arc length position and the chord position which are all given in dimensional values. All values are finally saved in a `.mat` file which is then used as in input for the transition prediction.

Nomenclature

In the nomenclature given here the superscript * is dropped and all quantities are given in dimensional form.

Latin Upper Case Letters

Symbol	Description	Units
$\boldsymbol{A}$	matrix for stability analysis	-
$\boldsymbol{A}_{(j)}^{(i)}$	matrix for boundary-layer calculation for $j = 0, 1, ..., J$ and $i = 0, 1, ...$	-
A	damping length constant for turbulence model	m
A	amplification factor for disturbances	-
A_0	amplification factor for disturbances at starting position	-
$\boldsymbol{B}$	matrix for stability analysis	-
$\boldsymbol{B}_{(j)}^{(i)}$	matrix for boundary-layer calculation for $j = 1, 2, ..., J$ and $i = 0, 1, ...$	-
$\boldsymbol{C}$	matrix for stability analysis	-
$\boldsymbol{C}_{(j)}^{(i)}$	matrix for boundary-layer calculation for $j = 0, 1, ..., J-1$ and $i = 0, 1, ...$	-
C	constant in intermittency factor for transition	-
C	summary of variables for stability analysis	-
C_f	skin friction coefficient	-
C_{ij}	coefficient to compute perturbation of the edge velocity	-
C_p	dimensionless pressure coefficient	-
$\boldsymbol{D}$	matrix for stability analysis	-
$\boldsymbol{D_1}$	differentiation matrix for first derivative	-
$D_{i,j}$	coefficient in differentiation matrices	-
D_j	coefficient to compute perturbation of the edge velocity	-
E_j^i	coefficients to compute perturbation of the edge velocity	-
F	dimensionless stream function f that also depends on s for similarity solution	-
F	dimensionless frequency	-
$F(\Gamma)$	function in second Walz method	-
$F(\lambda)$	function in first Walz method and Thwaites' mehtod	-
$F(\xi, \eta)$	dimensionless stream function for Görtler transformation	-
$F(x, Y)$	dimensionless stream function for inverse boundary-layer calculation	-
G	constant in intermittency factor for transition	-
$G(\lambda)$	subfunction of $F(\lambda)$ in first Walz method and Thwaites' method	-
H_{12}	shape factor $H_{12} = \frac{\delta_1}{\delta_2}$	-
J	total number of grid points in wall-normal direction	-
K	variable grid spacing parameter	-
$\boldsymbol{L}$	linear operator for stability analysis	-
$\boldsymbol{L}$	left-hand side matrix to solve direct Falkner-Skan transformation	-

Symbol	Description	Units
$\boldsymbol{L}_{ij}$	submatrices of linear operator for transition prediction for $i = 1,2,3,4$ and $j = 1,2,3,4$	-
$L_{(j-\frac{1}{2})}^{(n-1)}$	constant including all terms of the previous position of left-hand side for boundary-layer transformations for $j = 1,2,...,J$	-
L_c	chord length	m
L_{ref}	reference length	m
$\boldsymbol{M}$	diagonal matrix containing linear mapping function on its trace (for Chebyshev differentiation matrix)	-
Ma	Mach number	-
N	constant in damping length constant for turbulence model	-
N	total number of grid points in streamwise direction	-
N	total number of collocation points for Chebyshev polynomials	-
N	N-factor (amplification factor of disturbances)	-
N_E	N-factor (amplification factor of disturbances) based on energy	-
N_{trans}	critical N-factor when transition starts	-
P	pressure of the mean flow	$\mathrm{kg/m\,s^2}$
$R_{(j-\frac{1}{2})}^{(n-1)}$	constant including all terms of previous position of the right-hand side for boundary-layer transformations for $j = 1,2,...,J$	-
R	specific gas constant	$\mathrm{m^2/s^2\,K}$
Re	Reynolds number	-
Re_{δ_2}	Reynolds number based on the momentum thickness as length scale	-
T_∞	free-stream temperature	K
$\vec{U}$	velocity vector of mean flow $\vec{U} = (U,V,W)^T$	m/s
U	streamwise velocity of mean flow	m/s
U	first derivative of dimensionless stream function f with respect to s	-
U	first derivative of dimensionless stream function F in wall-normal direction η	-
U_e	velocity at boundary-layer edge	m/s
U_e^0	velocity at boundary-layer edge from inviscid solver	m/s
$U_{e,i}$	velocity at boundary-layer edge from inviscid solver	m/s
$U_{e,v}$	velocity at boundary-layer edge from viscous solver	m/s
U_{ref}	reference velocity	m/s
U_∞	free-stream velocity	m/s
V	wall-normal velocity of mean flow	m/s
V	second derivative of dimensionless stream function F in wall-normal direction η	-
V	second derivative of dimensionless stream function f with respect to s	-
$\boldsymbol{W}$	Chebyshev integral weight matrix	-
W	spanwise velocity of mean flow	m/s
Y	dimensionless wall-normal distance for inverse boundary-layer calculation	-
Y_{max}	maximum dimensionless wall-normal distance for inverse boundary-layer calculation	-
Z	function in first Walz method	s

Symbol	Description	Units
Z	function in second Walz method	m

Latin Lower Case Letters

Symbol	Description	Units
a	coefficient for Pohlhausen method	-
a	constant for first Walz method	-
a	constant for second Walz method	-
a	constant in stream function for a wedge flow	-
a	lower limit of physical domain for Chebyshev polynomials	-
$\boldsymbol{b}$	right-hand side matrix to solve direct Falkner-Skan transformation	-
b	coefficient for Pohlhausen method	-
b	constant for first Walz method	-
b	constant for second Walz method	-
b	dimensionless viscosity including turbulent viscosity	-
b	upper limit of physical domain for Chebyshev polynomials	-
b_j	Chebyshev constants	-
c	coefficient for Pohlhausen method	-
c_n	Chebyshev constants	-
c_∞	speed of sound	m/s
d	coefficient for Pohlhausen method	-
$d_{n,1}$	distance between two grid points to find Taylor series expansion for $n = 1, 2, ..., N$	-
$d_{n,2}$	distance between two grid points to find Taylor series expansion for $n = 1, 2, ..., N$	-
e	coefficient for Pohlhausen method	-
e_j	coefficient for Thomas algorithm for $j = 1, 2, 3$	-
$\vec{f}$	body force vector per unit volume	$\mathrm{kg/m^2\,s^2}$
f	frequency	1/s
f_{aux}	auxiliary function for PSE	-
$f(x, \eta)$	dimensionless stream function for Falkner-Skan transformation	-
$f(\eta)$	function for Pohlhausen's quartic velocity profile	-
$g(\Lambda)$	function in Pohlhausen differential equation	-
$\vec{g}$	gravitational acceleration	$\mathrm{m/s^2}$
g_i	coefficient to compute perturbation of the edge velocity	-
h	height	m
$h_{(j)}$	grid spacing in wall-normal direction between the two grid points j and $j-1$	-
h_1	grid spacing of first grid cell	-
$h(\Lambda)$	function in the Pohlhausen differential equation	-
i	iteration counter	-
j	grid position in wall-normal direction	-
k	wave number	1/m
$k^{(n)}$	grid spacing in streamwise direction between two grid points n and $n-1$	-
l	mixing length in turbulence model	m

Symbol	Description	Units
m	dimensionless pressure gradient in Falkner-Skan equation	-
m_1	slope of line to find stagnation point	-
m_2	slope of line to find stagnation point	-
n	constant for stream function of wedge flow	-
n	grid position in streamwise direction	-
p	pressure	$\mathrm{kg/m\,s^2}$
p'	pressure of the perturbation	$\mathrm{kg/m\,s^2}$
p^+	dimensionless pressure in turbulence model	-
$\hat{p}$	shape function of perturbation pressure	-
$\boldsymbol{q'}$	disturbance vector	-
$\hat{\boldsymbol{q}}$	shape functions of disturbances	-
$\hat{\boldsymbol{q}}_{\mathrm{init}}$	shape functions of initial disturbances	-
$\boldsymbol{r}$	right-hand side vector to solve direct Falkner-Skan transformation	-
$\vec{r}_{(j)}^{(i)}$	right-hand side vector of boundary-layer transformation for $j = 0, 1, ..., J$	-
r	independent coordinate in cylindrical coordinate system (to describe stream function of wedge flow)	-
r_j	coefficient of right-hand side of boundary-layer transformation for $j = 1, 2, 3$	-
r_4	coefficient of right-hand side of boundary-layer transformation for inverse method	-
s	value of the second derivative of f at the wall for similarity flows	-
s_j	coefficient from boundary-layer transformation for $j = 1, 2, ..., 6$	-
s_j	coefficient from boundary-layer transformation for inverse method for $j = 7, 8$	-
t	time	s
$\boldsymbol{u}$	solution vector of direct Falkner-Skan transformation	-
$\vec{u}$	velocity vector $\vec{u} = (u, v, w)^T$	m/s
$\vec{u}'$	velocity vector of perturbations $\vec{u}' = (u', v', w')^T$	m/s
u	streamwise velocity	m/s
u'	perturbations of streamwise velocity	m/s
u	first derivative of dimensionless stream function f in wall-normal direction η	-
$\hat{u}$	shape function of streamwise perturbation velocity	-
u_τ	friction velocity	m/s
v	wall-normal velocity	m/s
v	second derivative of dimensionless stream function f in wall-normal direction η	-
v'	perturbations of wall-normal velocity	m/s
$\hat{v}$	shape function of wall-normal perturbation velocity	-
v_0	blowing/suction velocity at wall	m/s
$\vec{w}_{(j)}$	vector for Thomas algorithm for $j = 0, 1, ..., J$	-
w	spanwise velocity	m/s
w'	perturbations of spanwise velocity	m/s

Symbol	Description	Units
$\hat{w}$	shape function of spanwise perturbation velocity	-
$\vec{x}$	airfoil coordinates $\vec{x} = (x, y, z)^T$	m
$\vec{x}_c$	profile coordinates $\vec{x}_c = (x_c, y_c)^T$	m
x	streamwise direction in airfoil coordinates	m
x_a	start of region where edge velocity has to be iterated	m
x_b	end of region where edge velocity has to be iterated	m
x_c	streamwise direction in profile coordinates	m
x_{stag}	arc length of stagnation point from trailing edge of airfoil	m
x_{tr}	position where transition starts	m
y	wall-normal direction in airfoil coordinates	m
Y_c	wall-normal direction in profile coordinates	m
y_c	position to switch from inner to outer turbulent viscosity	m
y_i	position where half of the grid points are located in nonlinear Chebyshev mapping	m
y_j	physical grid points of Chebyshev polynomials	-
$\hat{y}_j$	roots of Chebyshev polynomials	-
z	spanwise direction in airfoil coordinate	m

Greek Upper Case Letters

Symbol	Description	Units
$\mathbf{\Delta}_{(j)}$	matrix for Thomas algorithm for $j = 0, 1, ..., J$	-
$\mathbf{\Gamma}_{(j)}$	matrix for Thomas algorithm for $j = 1, 2, ..., J$	-
Γ	shape parameter in the second Walz method	-
Λ	Pohlhausen parameter (dimensionless pressure gradient parameter)	-
Λ_0	Pohlhausen parameter at the stagnation point	-
Ω'_y	perturbation vorticity in wall-normal direction	1/s
$\hat{\Omega}_y$	shape function of perturbation vorticity in wall-normal direction	-

Greek Lower Case Letters

Symbol	Description	Units
α	angle of attack	°
α	constant in turbulence viscosity for outer region $\alpha = 0.0168$	-
α	streamwise wave number	1/m
$\alpha^{(n)}$	constants for boundary-layer transformations	-
$\alpha_1^{(n)}$	constants for boundary-layer transformations	-
$\alpha_2^{(n)}$	constants for boundary-layer transformations	-
α_{ij}	coefficients of $\mathbf{\Delta}$ in Thomas algorithm for $i = 1, 2, 3$ and $j = 1, 2, 3$	-
β	wedge angle	°
β	spanwise wave number	1/m
β_H	Hartree parameter (dimensionless pressure gradient)	-
$\beta(\xi)$	principal function for Görtler transformation	-
$\vec{\delta}_{(j)}^{(i)}$	solution vector for boundary-layer transformation for $j = 0, 1, ..., J$	-

Symbol	Description	Units
δ	boundary-layer thickness	m
δ_1	displacement thickness	m
δ_1^0	displacement thickness of first iteration in inverse boundary-layer calculation	m
δ_2	momentum thickness	m
$\delta f_{(j)}^{(i)}$	iterate of stream function for Falkner-Skan transformation	-
$\delta F_{(j)}^{(i)}$	iterate of stream function for Görtler transformation	-
$\delta u_{(j)}^{(i)}$	iterate of first derivative of stream function for Falkner-Skan transformation	-
$\delta U_{(j)}^{(i)}$	iterate of first derivative of stream function for Görtler transformation	-
δU_e	small perturbation in the edge velocity	m/s
$\delta U_{e(j)}^{(i)}$	iterate of the edge velocity	-
$\delta v_{(j)}^{(i)}$	iterate of second derivative of stream function for Falkner-Skan transformation	-
$\delta V_{(j)}^{(i)}$	iterate of second derivative of stream function for Görtler transformation	-
$\boldsymbol{\phi}$	vector of dimensionless stream function and the derivatives $\boldsymbol{y} = (f, u, v)^T$	-
$\phi_{(j)}^{(n)}$	dummy variable on which substitutions are applied for Keller Box discretization	-
ϕ	independent coordinate in cylindrical coordinate system (to describe stream function of wedge flow)	-
γ	intermittency factor for outer region in turbulence model	-
γ	ratio of specific heats	-
γ_j	constants for inverse method for $j = 1, 2, 3$	-
γ_c	position to switch from inner to outer turbulent viscosity	m
γ_{ij}	coefficients of $\boldsymbol{\Gamma}$ in Thomas algorithm for $i = 1, 2, 3$ and $j = 1, 2, 3$	-
γ_{tr}	intermittency factor for transition	-
η	dimensionless wall-normal distance	-
η	wall-normal direction of grid	-
η_0	limit to calculate δ_1 when using the dimensionless stream function	-
$\eta_{99.5}$	dimensionless wall-normal distance where 99.5% of the free-stream velocity is reached	-
η_∞	dimensionless wall-normal distance at infinity	-
η_{max}	wall-normal distance at edge of grid	-
η_β	dimensionless wall-normal distance based on m	-
κ	constant in turbulence model $\kappa = 0.4$	-
λ	bulk viscosity $\lambda = -\frac{2}{3}\mu$	kg/m s
λ	dimensionless pressure gradient based on δ_2	-
λ	coefficient to compute the perturbation of the edge velocity	-
λ_0	dimensionless pressure gradient based on δ_2 at the stagnation point	-
λ_1	constant in inverse method	-
λ_2	constant in inverse method	-

Symbol	Description	Units
μ	dynamic viscosity	$\mathrm{kg/m\,s}$
ν	kinematic viscosity	$\mathrm{m^2/s}$
ν_t	turbulent kinematic viscosity	$\mathrm{m^2/s}$
$\nu_{t,i}$	turbulent kinematic viscosity for inner region of turbulent flow	$\mathrm{m^2/s}$
$\nu_{t,o}$	turbulent kinematic viscosity for outer region of turbulent flow	$\mathrm{m^2/s}$
ρ	density	$\mathrm{kg/m^3}$
$\rho\overline{u'v'}$	Reynolds shear stress	$\mathrm{kg/m\,s^2}$
σ	integration dummy variable	-
σ	growth rate	$\mathrm{1/m}$
σ_E	growth rate based on energy	$\mathrm{1/m}$
$\boldsymbol{\tau}$	shear stress tensor	$\mathrm{kg/m\,s^2}$
τ	shear stress	$\mathrm{kg/m\,s^2}$
τ_w	shear stress at wall	$\mathrm{kg/m\,s^2}$
ω	relaxation factor	-
ω	angular frequency	$\mathrm{1/s}$
ψ	stream function	$\mathrm{m^2/s}$
ψ_β	stream function based on dimensionless pressure gradient m	$\mathrm{m^2/s}$
ξ	streamwise direction of numerical grid	-
ξ	dimensionless streamwise coordinate for Görtler transformation	-

List of Figures

1.1. The shape of the *MW-airfoil* and characteristic flow values. 2

2.1. Coordinate system in airfoil flows. 8
2.2. The shape of the velocity profiles as a function of the Pohlhausen parameter Λ. The stagnation point and the separation point profiles are given for $\Lambda = 7.0523$ and $\Lambda = -12$, respectively. 11
2.3. The universal function $F(\lambda)$ needed to integrate the differential equation (2.46) and the approximation to find the approximate solution given in equation (2.52). 14
2.4. The universal function $F(\Gamma)$ needed to integrate the differential equation (2.53) and the approximations to find the approximate solution given in equation (2.52) with the two different values based on Walz [26] and Schlichting [20]. 16
2.5. Comparison between physical and dimensionless computation domain based on the numerical grid. 18
2.6. The dimensionless pressure gradient m can be used to define the wedge angle β. 21
2.7. Triple-Deck. 28

3.1. The numerical grid with the characteristics of a arbitrarily spacing in ξ-direction and a non-uniform spacing defined by the grid parameter K in η-direction. 36
3.2. Effect of grid parameters. 37
3.3. Extrapolation of the stagnation point. 38
3.4. The arc length of the stagnation point is located where the two straight lines cross. 39
3.5. Sketches for the derivation of the differencing schemes. 42
3.6. Boundary-layer characteristics of the suction side for the Walz method (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 46
3.7. Boundary-layer characteristics of the suction side for the Thwaites method (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 47
3.8. Falkner-Skan solutions for the Blasius case ($\beta_H = 0$). 49
3.9. Falkner-Skan solutions for different Hartree parameters. 50
3.10. Boundary-layer characteristics of the pressure side for the Falkner-Skan transformation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 54
3.11. Boundary-layer characteristics of the suction side for the Falkner-Skan transformation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 55
3.12. Boundary-layer profiles of the pressure side at $x_c = 0.0594$ for the Falkner-Skan transformation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 57
3.13. Boundary-layer profiles of the suction side at $x_c = 0.0373$ for the Falkner-Skan transformation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 58
3.14. Sketches for the derivation of the differencing schemes using the Keller box. 59
3.15. Boundary-layer characteristics of the pressure side for the Falkner-Skan transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 64

3.16. Boundary-layer characteristics of the suction side for the Falkner-Skan transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 65
3.17. Boundary-layer profiles of the pressure side at $x_c = 0.0594$ for the Falkner-Skan transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 67
3.18. Boundary-layer profiles of the suction side at $x_c = 0.0373$ for the Falkner-Skan transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 68
3.19. Boundary-layer characteristics of the pressure side for the Görtler transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 72
3.20. Boundary-layer characteristics of the suction side for the Görtler transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 73
3.21. Boundary-layer profiles of the pressure side at $x_c = 0.0594$ for the Görtler transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 75
3.22. Boundary-layer profiles of the suction side at $x_c = 0.0373$ for the Görtler transformation using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 76
3.23. Comparison of the boundary-layer profiles of the suction side at $x_c = 0.0373$ (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 78

4.1. Boundary-layer characteristics of the pressure side for the Falkner-Skan transformation with turbulence model (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 81
4.2. Boundary-layer characteristics of the suction side for the Falkner-Skan transformation with turbulence model (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 82
4.3. Boundary-layer characteristics of the pressure side for the Falkner-Skan transformation with turbulence model using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 84
4.4. Boundary-layer characteristics of the suction side for the Falkner-Skan transformation with turbulence model using the Keller box (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 85
4.5. Boundary-layer characteristics of the pressure side for the inverse method (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 90
4.6. Boundary-layer characteristics of the suction side for the inverse method (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 91

5.1. Comparison between Chebyshev and physical domain for different mapping functions. . . 93
5.2. The eigenvalues α are plotted for two different numbers of collocation points to find an initial guess for the Tollmien-Schlichting mode. 95
5.3. Growth rate for a Blasius flow with $F = 150$. 97
5.4. Neutral stability curves for different Hartree parameters. 97
5.5. N-factor for the pressure side (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 98
5.6. N-factor for the suction side (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 99
5.7. N-factor for different Hartree parameters using the formula of Drela and Giles [11]. 100
5.8. Comparison of N-factors between empirical method of Drela and Giles [11] and explicit numerical computation (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 101
5.9. Comparison between Orr-Sommerfeld and PSE solver for the neutral stability curve of a Blasius flow. 103
5.10. N-factor for the pressure side (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 104
5.11. N-factor for the suction side (*MW-airfoil*, $\alpha = 3.3°$, $Re = 2\,351\,842$). 104

Literature

[1] J.D. Anderson. Ludwig Prandtl's Boundary Layer. *Physics Today*, 58:42–48, 2005.

[2] J.D. Anderson. *Fundamentals of Aerodynamics*. McGraw-Hill Higher Education, Boston [u.a.], 2007.

[3] N.S. Asaithambi. A numerical method for the solution of the Falkner-Skan equation. *Applied Mathematics and Computation*, 81(2-3):259–264, Feb 1997.

[4] D. Catherall and K.W. Mangler. The integration of the Two-Dimensional laminar Boundary-Layer equations past the point of vanishing skin friction. *Journal of Fluid Mechanics*, 26(01):163–182, 1966.

[5] T. Cebeci. *An Engineering Approach to the Calculation of Aerodynamic Flows*. Springer, Berlin, Germany, 1999.

[6] T. Cebeci. *Turbulence Models and Their Application*. Springer, 2004.

[7] T. Cebeci. *Analysis of Low-Speed Unsteady Airfoil Flows*. Springer, April 2005.

[8] T. Cebeci and J. Cousteix. *Modeling and Computation of Boundary-Layer Flows*. Springer, Berlin, Germany, 2nd edition, 2005.

[9] T. Cebeci and H.B. Keller. Shooting and Parallel Shooting Methods for Solving the Falkner-Skan Boundary-Layer Equation. *Journal of Computational Physics*, 7(2):289–300, 1971.

[10] T. Cebeci, J. Shao, F. Kafyeke, and E. Laurendeau. *Computational Fluid Dynamics for Engineers*. Springer, Berlin, Germany, 2005.

[11] M. Drela and M.B. Giles. Viscous-Inviscid Analysis of Transonic and Low Reynolds Number Airfoils. *AIAA Journal*, 25(10):1347–1355, OCT 1987.

[12] S. Goldstein. On Laminar Boundary-Layer Flow near a Position of Separation. *The Quarterly Journal of Mechanics and Applied Mathematics*, 1(1):43 –69, January 1948.

[13] H. Görtler. A New Series for the Calculation of Steady Laminar Boundary Layer Flows. *Journal of Mathematics and Mechanics*, 6(1):1–66, 1957.

[14] A. Hanifi, O. Amoignon, J.O. Pralits, and M. Chevalier. A Gradient-based Optimization Method for Natural Laminar Flow Design. In Philipp Schlatter and Dan S. Henningson, editors, *Seventh IUTAM Symposium on Laminar-Turbulent Transition*, volume 18, pages 3–10. Springer Netherlands, Dordrecht, 2010.

[15] T. Herbert. Parabolized Stability Equations. *Annual Review of Fluid Mechanics*, 29(1):245–283, January 1997.

[16] K. Pohlhausen. Zur näherungsweisen Integration der Differentialgleichung der laminaren Grenzschicht. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 1(4):252–290, January 1921.

[17] A. Reeh. Development and Implementation of a Method for Linear Stability Analysis in Natural and Manipulated Boundary-Layer Flows. Technical report, TU Darmstadt, 2008.

[18] A. Reeh. In-Flight Investigation of Tollmien-Schlichting Waves Generated through Natural Boundary Layer Receptivity Mechanisms. Master's thesis, TU Darmstadt, 2009.

[19] H. Schlichting. *Grenzschicht-Theorie*. G. Braun, Karlsruhe, Germany, 8th edition, 1982.

[20] H. Schlichting and K. Gersten. *Boundary Layer Theory*. Springer, Berlin, Germany, 8th edition, 2000.

[21] P.J. Schmid and D.S. Henningson. *Stability and Transition in Shear Flows*. Springer, New York, first edition, 2001.

[22] J.H. Spurk and N. Aksel. *Strömungslehre - Einführung in die Theorie der Strömungen*. Springer, Berlin, Germany, 6th edition, 2006.

[23] D. Tempelmann, A. Hanifi, and D.S. Henningson. Spatial optimal growth in three-dimensional boundary layers. *Journal of Fluid Mechanics*, 646:5–37, 2010.

[24] J.L. Van Ingen. The en method for transition prediction. Historical review of work at TU Delft. In *38th AIAA Fluid Dynamics Conference and Exhibit, Seattle, USA, 23-26 June 2008*. AIAA, jan 2008.

[25] A. E. P. Veldman. New, Quasi-simultaneous Method to Calculate Interacting Boundary Layers. *AIAA Journal*, 19(1):79–85, January 1981.

[26] A. Walz. *Strömungs- und Temperaturgrenzschichten*. G. Braun, 1966.

[27] J.A. Weideman and S.C. Reddy. A MATLAB Differentiation Matrix Suite. *ACM - Transactions on Mathematical Software*, 26(4):465–519, Dec 2000.

[28] M. Weismüller, W. Friedrichs, A. Reeh, and C. Tropea. In-Flight Experiments on Natural Laminar Flow Airfoils under the Influence of Atmospheric Turbulence. In *48th AIAA Aerospace Sciences Meeting and Exhibit*, 2010.

[29] J. Zhang and B. Chen. An iterative method for solving the Falkner-Skan equation. *Applied Mathematics and Computation*, 210(1):215–222, Apr 2009.